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

In your requests please include the following reference number 78EE117C99-1289108940

Contents

1.1 checker-qual 2.11.1
   1.1.1 Available under license
1.2 open-csv 2.0
   1.2.1 Available under license
1.3 openjdk-jre 1.8.0u275
   1.3.1 Available under license
1.4 log4j-api 2.17.0
   1.4.1 Available under license
1.5 audit4j-spring 2.5.0
   1.5.1 Available under license
1.6 shadow 4.5-1ubuntu2.2
   1.6.1 Available under license
1.7 nettle 3.4.1-0ubuntu0.18.04.1
   1.7.1 Available under license
1.8 spring-framework 4.3.22.RELEASE
   1.8.1 Available under license
1.9 findutils 4.6.0+git+20170828 2
   1.9.1 Available under license
1.10 jackson-databind 2.10.0
   1.10.1 Available under license
1.11 zjsonpatch 0.3.0
   1.11.1 Available under license
1.12 moshi 1.8.0
   1.12.1 Available under license
1.13 metrics-servlet 4.1.9
   1.13.1 Available under license
1.14 libtasn 4.13 2
1.14.1 Available under license

1.15 nspr 4.18 1ubuntu1
1.15.1 Available under license

1.16 commons-logging 1.1.3
1.16.1 Available under license

1.17 kubernetes-model-flowcontrol 5.4.1
1.17.1 Available under license

1.18 pcre 8.39 9
1.18.1 Available under license

1.19 joda-time 2.8.2
1.19.1 Available under license

1.20 file 5.32-2ubuntu0.4
1.20.1 Available under license

1.21 metrics-jvm 4.1.9
1.21.1 Available under license

1.22 commons-lang3 3.5
1.22.1 Available under license

1.23 httpcore5 5.0
1.23.1 Available under license

1.24 javax-persistence-api 2.2
1.24.1 Available under license

1.25 qemu-utils 2.11+dfsg-1ubuntu7.39
1.25.1 Available under license

1.26 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava
1.26.1 Available under license

1.27 libhcryptoheimdal 7.5.0+dfsg-1
1.27.1 Available under license

1.28 fdisk 2.31.1
1.28.1 Available under license

1.29 gmp 6.1.2+dfsg-2
1.29.1 Available under license

1.30 awaitility 4.0.1
1.30.1 Available under license

1.31 libhcryptoheimdal 7.5.0+dfsg-1
1.31.1 Available under license

1.32 docker-java 3.2.5
1.32.1 Available under license

1.33 libaio 0.3.110 5ubuntu0.1
| 1.33.1 | Available under license |
| 1.34 | libseccomp 2.5.1-1ubuntu1~18.04.2 |
| 1.34.1 | Available under license |
| 1.35 | berkeley-db 5.3.28-13.1ubuntu1.1 |
| 1.35.1 | Available under license |
| 1.36 | dnsjava 2.1.7 |
| 1.36.1 | Available under license |
| 1.37 | groovy 2.1.5 |
| 1.37.1 | Available under license |
| 1.38 | sigar 1.6.5.132-5 |
| 1.38.1 | Available under license |
| 1.39 | libheimbaseheimdal 7.5.0+dfsg-1 |
| 1.39.1 | Available under license |
| 1.40 | netty-codec-socks 4.1.49.Final |
| 1.40.1 | Available under license |
| 1.41 | javax-el 3.0.0 |
| 1.41.1 | Available under license |
| 1.42 | junixsocket-native-common 2.3.2 |
| 1.42.1 | Available under license |
| 1.43 | javax-interceptor-api 1.2 |
| 1.43.1 | Available under license |
| 1.44 | hibernate-validator 6.1.6.Final |
| 1.44.1 | Available under license |
| 1.45 | libgssapiheimdal 7.5.0+dfsg-1 |
| 1.45.1 | Available under license |
| 1.46 | libhxheimdal 7.5.0+dfsg-1 |
| 1.46.1 | Available under license |
| 1.47 | dpkg 1.19.0.5ubuntu2.3 |
| 1.47.1 | Available under license |
| 1.48 | qemu-block-extra 2.11+dfsg-1ubuntu7.39 |
| 1.48.1 | Available under license |
| 1.49 | commons-configuration 1.5 |
| 1.49.1 | Available under license |
| 1.50 | xerces-j 2.7.1 |
| 1.50.1 | Available under license |
| 1.51 | spring-tx 5.2.9.RELEASE |
| 1.51.1 | Available under license |
| 1.52 | shiro-crypto-cipher 1.7.1 |
| 1.52.1 | Available under license |
1.53 ubuntu-keyring 2018.09.18.1~18.04.2
  1.53.1 Available under license
1.54 jetty-plus 9.4.31.v20200723
  1.54.1 Available under license
1.55 hdrhistogram 2.1.12
  1.55.1 Available under license
1.56 libsemanage 2.7 2build2
  1.56.1 Available under license
1.57 jetty-io 9.4.31.v20200723
  1.57.1 Available under license
1.58 httpcomponents-mime 4.5.13
  1.58.1 Available under license
1.59 util-linux 2.31.1 0.4ubuntu3.7
  1.59.1 Available under license
1.60 jsr166e 1.0
  1.60.1 Available under license
1.61 commons-io 2.4
  1.61.1 Available under license
1.62 spring-data-jpa 2.1.5.RELEASE
  1.62.1 Available under license
1.63 netty-transport-native- unix-common 4.1.46.Final
  1.63.1 Available under license
1.64 dash 0.5.8 2.10
  1.64.1 Available under license
1.65 log4j-jcl 2.17.0
  1.65.1 Available under license
1.66 metrics-annotation 3.1.2
  1.66.1 Available under license
1.67 lz4 0.0~r131-2ubuntu3.1
  1.67.1 Available under license
1.68 shiro-config-ogdl 1.7.1
  1.68.1 Available under license
1.69 libcap-ng 0.7.7 3.1
  1.69.1 Available under license
1.70 mockito 1.10.19
  1.70.1 Available under license
1.71 cglib 3.3.0
  1.71.1 Available under license
1.72 gson 2.8.0
1.72.1 Available under license
1.73 udev 241-7~deb10u3
  1.73.1 Available under license
1.74 hazelcast-kubernetes 1.0.0
  1.74.1 Available under license
1.75 activation-api 1.2.0
  1.75.1 Available under license
1.76 annotations 13.0
1.77 libkrbheimald 7.5.0+dfsg-1
  1.77.1 Available under license
1.78 stax-ex 1.8
  1.78.1 Available under license
1.79 libunistring 0.9.9 0ubuntu2
  1.79.1 Available under license
1.80 quartz 2.3.2
  1.80.1 Available under license
1.81 jcl-over-slf4j 1.7.30
  1.81.1 Available under license
1.82 bean-validation-api 2.0.1.Final
  1.82.1 Available under license
1.83 jackson-annotations 2.12.1
  1.83.1 Available under license
1.84 hazelcast 3.12.3
  1.84.1 Available under license
1.85 jersey-client 2.30.1
  1.85.1 Available under license
1.86 error_prone_annotations 2.3.4
  1.86.1 Available under license
1.87 mapdb 1.0.6
  1.87.1 Available under license
1.88 p11-kit 0.23.9-2ubuntu0.1
  1.88.1 Available under license
1.89 commons-dbcp 1.4
  1.89.1 Available under license
1.90 jetty-util 9.4.31.v20200723
  1.90.1 Available under license
1.91 kubernetes-model-common 5.4.1
  1.91.1 Available under license
1.92 selinux 2.7-2build2
1.92.1 Available under license
1.93 libnspr 4.18-1ubuntu1
1.93.1 Available under license
1.94 netty-handler 4.1.49.Final
1.94.1 Available under license
1.95 ncurses 6.1-1ubuntu1.18.04
1.95.1 Available under license
1.96 ini4j 0.5.4
1.96.1 Available under license
1.97 libidn 2.0.4-1.1ubuntu0.2
1.97.1 Available under license
1.98 dom4j 2.1.3
1.98.1 Available under license
1.99 commons-digester 2.1
1.99.1 Available under license
1.100 zlib 1.2.11.dfsg-0ubuntu2
1.100.1 Available under license
1.101 jradius-dictionary 1.1.6-SNAPSHOT
1.101.1 Available under license
1.102 libffi 3.2.1 8
1.102.1 Available under license
1.103 shiro 1.7.1
1.103.1 Available under license
1.104 jetty-webapp 9.4.31.v20200723
1.104.1 Available under license
1.105 librbd 12.2.13-0ubuntu0.18.04.10
1.105.1 Available under license
1.106 open-ldap 2.4.45+dfsg-1ubuntu1.10
1.106.1 Available under license
1.107 hibernate 5.2.17.Final
1.107.1 Available under license
1.108 libpwquality 1.4.0-2
1.108.1 Available under license
1.109 iconv 2.27
1.109.1 Available under license
1.110 jakarta-ws-rs-api 2.1.6
1.110.1 Available under license
1.111 kubernetes-model-node 5.4.1
1.111.1 Available under license
1.112 jakarta-inject 2.6.1  
   1.112.1 Available under license
1.113 libpng 1.6.37  
   1.113.1 Available under license
1.114 hk2-locator 2.6.1  
   1.114.1 Available under license
1.115 kubernetes-client 5.4.1  
   1.115.1 Available under license
1.116 giflib 5.2.1  
   1.116.1 Available under license
1.117 netty-handler-proxy 4.1.49.Final  
   1.117.1 Available under license
1.118 jackson-dataformat-yaml 2.11.2  
   1.118.1 Available under license
1.119 spring-jdbc 2.5.6  
   1.119.1 Available under license
1.120 shiro-spring 1.7.1  
   1.120.1 Available under license
1.121 saaj 1.5.2  
   1.121.1 Available under license
1.122 spring-aspects 5.2.9.RELEASE  
   1.122.1 Available under license
1.123 shiro-crypto-hash 1.7.1  
   1.123.1 Available under license
1.124 vertx-unit 3.9.4  
   1.124.1 Available under license
1.125 activation 1.2.0  
   1.125.1 Available under license
1.126 metrics-json 4.1.9  
   1.126.1 Available under license
1.127 pcre 8.39-9  
   1.127.1 Available under license
1.128 ejb-api 3.0-alpha-1  
   1.128.1 Available under license
1.129 perl 5.26.1 6ubuntu0.5  
   1.129.1 Available under license
1.130 libffi 3.2.1 8  
   1.130.1 Available under license
1.131 libsigar 1.6.5
1.131.1 Available under license
1.132 xstream 1.4.17
   1.132.1 Available under license
1.133 antlr 2.7.7
   1.133.1 Available under license
1.134 jsr305 3.0.2
   1.134.1 Available under license
1.135 swagger-annotations 1.5.20
1.136 jaxb-api 2.3.1
   1.136.1 Available under license
1.137 xml-commons-resolver 1.2
   1.137.1 Available under license
1.138 jta 1.3
   1.138.1 Available under license
1.139 libnss 3.35-2ubuntu2.13
   1.139.1 Available under license
1.140 management-api 1.1-rev-1
   1.140.1 Available under license
1.141 metrics-spring 3.1.3
   1.141.1 Available under license
1.142 fastjson 1.2.26
   1.142.1 Available under license
1.143 commons-text 1.9
   1.143.1 Available under license
1.144 profiler 1.1.1
   1.144.1 Available under license
1.145 libibverbs 17.1-1ubuntu0.2
   1.145.1 Available under license
1.146 flex-json 2.1
   1.146.1 Available under license
1.147 spring-beans 4.3.22.RELEASE
   1.147.1 Available under license
1.148 simpleclient-common 0.8.1
   1.148.1 Available under license
1.149 jdk-zulu 8.50.0.51
   1.149.1 Available under license
1.150 commons-beanutils 1.8.3
   1.150.1 Available under license
1.151 netty-resolver-dns 4.1.49.Final
1.151.1 Available under license
1.152 sysv-init 2.88dsf 59.10ubuntu1
1.152.1 Available under license
1.153 shiro-cache 1.7.1
1.153.1 Available under license
1.154 base-passwd 3.5.44
1.154.1 Available under license
1.155 commons-collections 3.2.2
1.155.1 Available under license
1.156 netty-buffer 4.1.49.Final
1.156.1 Available under license
1.157 vertx-web 3.9.4
1.157.1 Available under license
1.158 jandex 2.1.3.Final
1.158.1 Available under license
1.159 guava 29.0-jre
1.159.1 Available under license
1.160 spring-web-mvc 4.3.22.RELEASE
1.160.1 Available under license
1.161 diffutils 3.6-1
1.161.1 Available under license
1.162 spring-aop 4.3.22.RELEASE
1.162.1 Available under license
1.163 netty-transport 4.1.49.Final
1.163.1 Available under license
1.164 fast-infoset 1.2.15
1.165 sed 4.4 2
1.165.1 Available under license
1.166 tdb 1.44.1
1.166.1 Available under license
1.167 jakarta-validation-api 2.0.2
1.167.1 Available under license
1.168 javax-annotation-api 1.3
1.168.1 Available under license
1.169 netty-resolver 4.1.49.Final
1.169.1 Available under license
1.170 metrics-servlets 4.1.9
1.170.1 Available under license
1.171 ceph 12.2.13-0ubuntu0.18.04.10
1.171.1 Available under license
1.172 **classmate 1.5.1**
   1.172.1 Available under license
1.173 **cpp 8.4.0 1ubuntu1~18.04**
   1.173.1 Available under license
1.174 **gnutls 3.5.18-1ubuntu1.5**
   1.174.1 Available under license
1.175 **jetty-jndi 9.4.31.v20200723**
   1.175.1 Available under license
1.176 **latencyutils 2.0.3**
1.177 **debianutils 4.8.4**
   1.177.1 Available under license
1.178 **glib 2.56.4-0ubuntu0.18.04.9**
   1.178.1 Available under license
1.179 **heimdal 7.5.0+dfsg-1**
   1.179.1 Available under license
1.180 **jetty-security 9.4.31.v20200723**
   1.180.1 Available under license
1.181 **audit4j-core 2.6.0**
   1.181.1 Available under license
1.182 **vertx-bridge-common 3.9.4**
   1.182.1 Available under license
1.183 **curl 7.58.0 2ubuntu3.16**
   1.183.1 Available under license
1.184 **glibc 2.27-3ubuntu1.5**
   1.184.1 Available under license
1.185 **netty-transport-native-epoll 4.1.46.Final**
   1.185.1 Available under license
1.186 **libsigar 1.6.6**
   1.186.1 Available under license
1.187 **failureaccess 1.0.1**
   1.187.1 Available under license
1.188 **jakarta-annotation-api 1.3.5**
   1.188.1 Available under license
1.189 **jboss-transaction-api 1.1.1.Final**
1.190 **commons-pool 1.5.6**
   1.190.1 Available under license
1.191 **cdebconf 0.213ubuntu1**
   1.191.1 Available under license
<table>
<thead>
<tr>
<th>Version</th>
<th>Description</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.192.1</td>
<td>vertx-auth-common 3.9.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.193.1</td>
<td>swagger 1.5.20</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.194.1</td>
<td>influxdb-java 2.20</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.195.1</td>
<td>kerberos 1.16-2ubuntu0.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.196.1</td>
<td>libgcrypt 1.8.1 4ubuntu1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.197.1</td>
<td>openjssse 1.1.5</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.198.1</td>
<td>cyrus-sasl 2.1.27~101-g0780600+dfsg-3ubuntu2.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.199.1</td>
<td>netty-transport-native-kqueue 4.1.46.Final</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.200.1</td>
<td>jackson-annotations 2.10.0</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.201.1</td>
<td>httpcomponents-client 4.5.12</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.202.1</td>
<td>nss 3.35-2ubuntu2.13</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.203.1</td>
<td>flyway-core 4.1.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.204.1</td>
<td>cron-utils 9.1.3</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.205.1</td>
<td>grep 3.1 2build1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.206.1</td>
<td>jaxb-core 2.3.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.207.1</td>
<td>shiro-lang 1.7.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.208.1</td>
<td>gnupg 2.2.4 1ubuntu1.4</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.209.1</td>
<td>commons-logging 1.2</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.210.1</td>
<td>audit 2.8.2-1ubuntu1.1</td>
<td>Available under license</td>
</tr>
<tr>
<td>1.211.1</td>
<td>libsigar 1.6.4</td>
<td>Available under license</td>
</tr>
</tbody>
</table>
1.212 threetenbp 1.4.4  
   1.212.1 Available under license
1.213 slf4j-log4j 1.5.8  
   1.213.1 Available under license
1.214 bouncy-castle 1.64  
   1.214.1 Available under license
1.215 hamcrest 1.3  
   1.215.1 Available under license
1.216 okio 2.7.0  
   1.216.1 Available under license
1.217 shiro-event 1.7.1  
   1.217.1 Available under license
1.218 jackson-datatype-jsr310 2.11.2  
   1.218.1 Available under license
1.219 zlib 1.2.11.dfsg 0ubuntu2  
   1.219.1 Available under license
1.220 metrics-health-checks 4.1.9  
   1.220.1 Available under license
1.221 cracklib 2.9.2-5build1  
   1.221.1 Available under license
1.222 jackson 2.10.0  
   1.222.1 Available under license
1.223 libheimntlmheimdal 7.5.0+dfsg-1  
   1.223.1 Available under license
1.224 shiro-aspectj 1.7.1  
   1.224.1 Available under license
1.225 spring-plugin-core 1.2.0.RELEASE  
   1.225.1 Available under license
1.226 legacy8ujsse 1.1.1  
   1.226.1 Available under license
1.227 hikaricp-java7 2.4.13  
1.228 slf4j 1.7.30  
   1.228.1 Available under license
1.229 javassist 3.19.0-GA  
   1.229.1 Available under license
1.230 commons-codec 1.5  
   1.230.1 Available under license
1.231 hk2-utils 2.6.1  
   1.231.1 Available under license
1.232 nghttp2 1.30.0-1ubuntu1
  1.232.1 Available under license
1.233 docker-java-transport-httpclient 3.2.5
  1.233.1 Available under license
1.234 yamlbeans 1.09
1.235 istack-commons 3.0.7
  1.235.1 Available under license
1.236 kotlin 1.3.70
  1.236.1 Available under license
1.237 e2fsprogs 1.44.1 1ubuntu1.3
  1.237.1 Available under license
1.238 jetty-servlet 9.4.31.v20200723
  1.238.1 Available under license
1.239 xmlsec-java 2.1.4
  1.239.1 Available under license
1.240 encoder 1.2.2
  1.240.1 Available under license
1.241 bash 4.4.18 2ubuntu1.2
  1.241.1 Available under license
1.242 ehcache 2.4.0
  1.242.1 Available under license
1.243 jctools-core 3.0.0
  1.243.1 Available under license
1.244 libwindheimdal 7.5.0+dfsg-1
  1.244.1 Available under license
1.245 jmte 4.0.0
  1.245.1 Available under license
1.246 sensible-utils 0.0.12
  1.246.1 Available under license
1.247 berkeley-db 5.3.28-13.1ubuntu1
  1.247.1 Available under license
1.248 txw2 2.3.1
  1.248.1 Available under license
1.249 go-runtime 1.15.3
  1.249.1 Available under license
1.250 kubernetes-model 5.4.1
  1.250.1 Available under license
1.251 hazelcast-client-protocol 1.8.0
  1.251.1 Available under license
1.252 base-files 10.1ubuntu2.11
   1.252.1 Available under license
1.253 vertx-hazelcast 3.9.4
   1.253.1 Available under license
1.254 mawk 1.3.3 17ubuntu3
   1.254.1 Available under license
1.255 jetty 9.4.31.v20200723
   1.255.1 Available under license
1.256 vertx-web-common 3.9.4
   1.256.1 Available under license
1.257 snake-yaml 1.27
   1.257.1 Available under license
1.258 c3p0 0.9.5.4
   1.258.1 Available under license
1.259 libnlroute 3.2.29-0ubuntu3
   1.259.1 Available under license
1.260 zstd 1.3.3+dfsg 2ubuntu1.2
   1.260.1 Available under license
1.261 librados 12.2.13-0ubuntu0.18.04.10
   1.261.1 Available under license
1.262 asm 3.3.1
   1.262.1 Available under license
1.263 procps 3.3.12-3ubuntu1.2
   1.263.1 Available under license
1.264 postgresql-jdbc 9.3-1103-jdbc41
   1.264.1 Available under license
1.265 jetty-http 9.4.31.v20200723
   1.265.1 Available under license
1.266 log4j-slf4j-impl 2.17.0
   1.266.1 Available under license
1.267 httpclient5 5.0
   1.267.1 Available under license
1.268 libnl3 3.2.29 0ubuntu3
   1.268.1 Available under license
1.269 zlib 1.2.11
   1.269.1 Available under license
1.270 lcms 2.10
   1.270.1 Available under license
1.271 jersey-apache-connector 2.30.1
1.271.1 Available under license
1.272 libgpg-error 1.27 6
   1.272.1 Available under license
1.273 libncurses 6.1-1ubuntu1.18.04
   1.273.1 Available under license
1.274 jackson-datatype-hibernate 2.12.1
   1.274.1 Available under license
1.275 mxparser 1.2.1
   1.275.1 Available under license
1.276 httpcomponents-core 4.4.5
   1.276.1 Available under license
1.277 spring-web 4.3.22.RELEASE
   1.277.1 Available under license
1.278 rtmpdump 2.4+20151223.gitfa8646d.1 1
   1.278.1 Available under license
1.279 netty-codec-http 4.1.49.Final
   1.279.1 Available under license
1.280 xz 5.2.2 1.3
   1.280.1 Available under license
1.281 libiscsi 1.17.0-1.1
   1.281.1 Available under license
1.282 librokenheimdal 7.5.0+dfsg-1
   1.282.1 Available under license
1.283 metrics 4.1.9
   1.283.1 Available under license
1.284 hk2-api 2.6.1
   1.284.1 Available under license
1.285 javax-enterprise-concurrent-api 1.0
   1.285.1 Available under license
1.286 netty-codec-dns 4.1.49.Final
   1.286.1 Available under license
1.287 jackson 2.1.0
   1.287.1 Available under license
1.288 jackson-databind 2.12.1
   1.288.1 Available under license
1.289 apt 1.6.14
   1.289.1 Available under license
1.290 jna 5.5.0
   1.290.1 Available under license
<table>
<thead>
<tr>
<th>Open Source Used In IOx-FogDirector-AC18 1.17.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.291 generex 1.0.2</td>
</tr>
<tr>
<td>1.291.1 Available under license</td>
</tr>
<tr>
<td>1.292 junixsocket-common 2.3.2</td>
</tr>
<tr>
<td>1.292.1 Available under license</td>
</tr>
<tr>
<td>1.293 objenesis 1.0</td>
</tr>
<tr>
<td>1.293.1 Available under license</td>
</tr>
<tr>
<td>1.294 docker-java-transport 3.2.5</td>
</tr>
<tr>
<td>1.294.1 Available under license</td>
</tr>
<tr>
<td>1.295 acl 2.2.52 3build1</td>
</tr>
<tr>
<td>1.295.1 Available under license</td>
</tr>
<tr>
<td>1.296 qemu 2.11+dfsg-1ubuntu7.39</td>
</tr>
<tr>
<td>1.296.1 Available under license</td>
</tr>
<tr>
<td>1.297 spring-context 4.3.22.RELEASE</td>
</tr>
<tr>
<td>1.297.1 Available under license</td>
</tr>
<tr>
<td>1.298 spring-expression 5.2.9.RELEASE</td>
</tr>
<tr>
<td>1.298.1 Available under license</td>
</tr>
<tr>
<td>1.299 byte-buddy 1.10.10</td>
</tr>
<tr>
<td>1.299.1 Available under license</td>
</tr>
<tr>
<td>1.300 sqlite 3.22.0-1ubuntu0.4</td>
</tr>
<tr>
<td>1.300.1 Available under license</td>
</tr>
<tr>
<td>1.301 libpsl 0.19.1-5build1</td>
</tr>
<tr>
<td>1.301.1 Available under license</td>
</tr>
<tr>
<td>1.302 spring-hateoas 0.25.1.RELEASE</td>
</tr>
<tr>
<td>1.302.1 Available under license</td>
</tr>
<tr>
<td>1.303 aspect-j 1.8.7</td>
</tr>
<tr>
<td>1.303.1 Available under license</td>
</tr>
<tr>
<td>1.304 hostname 3.20</td>
</tr>
<tr>
<td>1.304.1 Available under license</td>
</tr>
<tr>
<td>1.305 okhttp 4.8.1</td>
</tr>
<tr>
<td>1.305.1 Available under license</td>
</tr>
<tr>
<td>1.306 httpcomponents-core 4.4.13</td>
</tr>
<tr>
<td>1.306.1 Available under license</td>
</tr>
<tr>
<td>1.307 jradius-core 1.1.6-SNAPSHOT</td>
</tr>
<tr>
<td>1.307.1 Available under license</td>
</tr>
<tr>
<td>1.308 commons-fileupload 1.3.3</td>
</tr>
<tr>
<td>1.308.1 Available under license</td>
</tr>
<tr>
<td>1.309 debconf 1.5.66ubuntu1</td>
</tr>
<tr>
<td>1.310 keyutils 1.5.9 9.2ubuntu2</td>
</tr>
<tr>
<td>1.310.1 Available under license</td>
</tr>
</tbody>
</table>
1.311 lsb 9.20170808ubuntu1
  1.311.1 Available under license
1.312 gzip 1.6 5ubuntu1.1
  1.312.1 Available under license
1.313 resourcelocator 1.0.3
  1.313.1 Available under license
1.314 datasource-proxy 1.4.1
  1.314.1 Available under license
1.315 adduser 3.116ubuntu1
  1.315.1 Available under license
1.316 vertx-core 3.9.4
  1.316.1 Available under license
1.317 free-type 2.8.1
  1.317.1 Available under license
1.318 servlet-api 3.1.0
  1.318.1 Available under license
1.319 shiro-crypto-core 1.7.1
  1.319.1 Available under license
1.320 tar 1.29b 2ubuntu0.2
  1.320.1 Available under license
1.321 libaio 0.3.110-5ubuntu0.1
  1.321.1 Available under license
1.322 spring-data-commons 2.1.5.RELEASE
  1.322.1 Available under license
1.323 readline 8.0.4
  1.323.1 Available under license
1.324 spring-plugin-metadata 1.2.0.RELEASE
  1.324.1 Available under license
1.325 bzip2 1.0.6 8.1ubuntu0.2
  1.325.1 Available under license
1.326 jersey-hk2 2.30.1
  1.326.1 Available under license
1.327 shiro-config-core 1.7.1
  1.327.1 Available under license
1.328 rdma-core 17.1-1ubuntu0.2
  1.328.1 Available under license
1.329 netty 4.1.49.Final
  1.329.1 Available under license
1.330 systemd 237-3ubuntu10.53
1.330.1 Available under license
1.331 libiscsi 1.17.0-1.1
   1.331.1 Available under license
1.332 commons-chain 1.2
   1.332.1 Available under license
1.333 netty-codec 4.1.49.Final
   1.333.1 Available under license
1.334 jersey 2.30.1
   1.334.1 Available under license
1.335 shiro-quartz 1.7.1
   1.335.1 Available under license
1.336 commons-compress 1.19
   1.336.1 Available under license
1.337 kerberos 1.16-2ubuntu0.2
   1.337.1 Available under license
1.338 init-system-helpers 1.51
   1.338.1 Available under license
1.339 simpleclient 0.8.1
   1.339.1 Available under license
1.340 coreutils 8.28 1ubuntu1
   1.340.1 Available under license
1.341 commons-lang3 3.4
   1.341.1 Available under license
1.342 commons-codec 1.11
   1.342.1 Available under license
1.343 jul-to-slf4j 1.7.30
   1.343.1 Available under license
1.344 libsepol 2.7-1
   1.344.1 Available under license
1.345 mapstruct 1.2.0.Final
1.346 jetty-annotations 9.4.31.v20200723
   1.346.1 Available under license
1.347 libjpeg 6b
   1.347.1 Notifications
   1.347.2 Available under license
1.348 httpcomponents-client 4.5.13
   1.348.1 Available under license
1.349 shiro-web 1.7.1
   1.349.1 Available under license
1.350 j2objc-annotations 1.3
1.351 log4j-web 2.17.0
  1.351.1 Available under license
1.352 jetty-xml 9.4.31.v20200723
  1.352.1 Available under license
1.353 scowl 2017.08.24-1
  1.353.1 Available under license
1.354 libsigar 1.6.2
  1.354.1 Available under license
1.355 pam 1.1.8-3.6ubuntu2.18.04.3
  1.355.1 Available under license
1.356 apache-log4j 2.17.0
  1.356.1 Available under license
1.357 commons-net 3.4
  1.357.1 Available under license
1.358 jboss-logging 3.2.0.Final
  1.358.1 Available under license
1.359 attr 2.4.47-2build1
  1.359.1 Available under license

1.1 checker-qual 2.11.1
1.1.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**GNU GENERAL PUBLIC LICENSE**

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.
### 1.2 open-csv 2.0

#### 1.2.1 Available under license:

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

```
/**
 * Copyright 2005 Bytecode Pty Ltd.

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

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

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/
/**
 * A very simple CSV writer released under a commercial-friendly license.
 *
 * @author Glen Smith
 *
 */
```

Found in path(s):
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2005 Bytecode Pty Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * A very simple CSV reader released under a commercial-friendly license.
 *
 * @author Glen Smith
 *
 */

Found in path(s):
* /opt/cola/permits/1283432138_1646722275.6/0/opencsv-2-0-sources-jar/au/com/bytecode/opencsv/CSVReader.java

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

/**
 * Copyright 2007 Kyle Miller.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
A very simple CSV parser for Java released under a commercial-friendly license.

**Harfbuzz v2.7.2**

Harfbuzz is licensed under the so-called "Old MIT" license. Details follow. For parts of Harfbuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright 2018,2019,2020 Ebrahim Byagowi
Copyright 2019,2020 Facebook, Inc.
Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stribley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.
Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

All source code, except for one section, is licensed as above. The one exception is licensed with a slightly different MIT variant:
The contents of this directory are licensed under the following terms:

Copyright (C) 2012 Grigori Goronzy <greg@kinoho.net>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1999-2003 David Corcoran <corcoran@musclecard.com>
Copyright (c) 2001-2011 Ludovic Rousseau <ludovic.rousseau@free.fr>
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

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

Some files are under GNU GPL v3 or any later version
- doc/example/pcsc_demo.c
- the files in src/spy/
- the files in UnitaryTests/

Copyright (C) 2003-2014 Ludovic Rousseau

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Files src/auth.c and src/auth.h are:
* Copyright (C) 2013 Red Hat
* All rights reserved.
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
* FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
* COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
* BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
* OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
* AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
* THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
* DAMAGE.
*
* Author: Nikos Mavrogianopoulos <nmav@redhat.com>

Files src/simclist.c and src/simclist.h are:

* Copyright (c) 2007,2008,2009,2010,2011 Mij <mij@bitchx.it>

* Permission to use, copy, modify, and distribute this software for any
* purpose with or without fee is hereby granted, provided that the above
* copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
* ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
* ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
* OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
## c-libutl 20160225

### c-libutl License
```
This software is distributed under the terms of the BSD license.

== BSD LICENSE ==============================================================

(C) 2009 by Remo Dentato (rdentato@gmail.com)```
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

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

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

http://opensource.org/licenses/bsd-license.php

```
## xwd v1.0.7

### xwd utility
<pre>
This is the copyright for the files in src/java.desktop/unix/native/libawt_xawt:
list.h, multiVis.h, wsutils.h, list.c, multiVis.c

Copyright 1994 Hewlett-Packard Co.
Copyright 1996, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

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

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

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

</pre>
The GIFLIB distribution is Copyright (c) 1997 Eric S. Raymond

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

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

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

## IAIK (Institute for Applied Information Processing and Communication) PKCS#11 wrapper files v1

### IAIK License

Copyright (c) 2002 Graz University of Technology. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
"This product includes software developed by IAIK of Graz University of Technology."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Graz University of Technology" and "IAIK of Graz University of Technology" must not be used to endorse or promote products derived from this software without prior written permission.

5. Products derived from this software may not be called "IAIK PKCS Wrapper", nor may "IAIK" appear in their name, without prior written permission of Graz University of Technology.

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

</pre>

## zlib v1.2.11

### zlib License

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

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

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

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
FreeType comes with two licenses from which you can choose the one which fits your needs best.

The FreeType License (FTL) is the most commonly used one. It is a BSD-style license with a credit clause and thus compatible with the GNU Public License (GPL) version 3, but not with the GPL version 2.

The GNU General Public License (GPL), version 2. Use it for all projects which use the GPLv2 also, or which need a license compatible to the GPLv2.

The FreeType Project LICENSE
----------------------------
2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction
============

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license
affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be interested in any kind of bug reports. ("as is" distribution)

- You can use this software for whatever you want, in parts or full form, without having to pay us. ("royalty-free" usage)

- You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. ("credits")

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```markdown
---
Portions of this software are copyright <year> The FreeType Project (www.freetype.org). All rights reserved.
---

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms
-------------

0. Definitions
-------------

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha,
beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

-----------

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

-----------

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file (`FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that
states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: "FreeType Project", "FreeType Engine", "FreeType library", or "FreeType Distribution".

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

o freetype@nongnu.org

Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

o freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at
http://www.freetype.org

### GPL v2

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

   Gnomovision version 69, Copyright (C) year name of author
   Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
   This is free software, and you are welcome to redistribute it
   under certain conditions; type `show c' for details.

   The hypothetical commands `show w' and `show c' should show the appropriate
   parts of the General Public License. Of course, the commands you use may
   be called something other than `show w' and `show c'; they could even be
   mouse-clicks or menu items--whatever suits your program.

   You should also get your employer (if you work as a programmer) or your
   school, if any, to sign a "copyright disclaimer" for the program, if
   necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

#########################################################################
--- fthash.c and fthash.h are covered by the following notices ---

/ *
* Copyright 2000 Computing Research Labs, New Mexico State University
* Copyright 2001-2015
* Francesco Zappa Nardelli
* *
* Permission is hereby granted, free of charge, to any person obtaining a
* copy of this software and associated documentation files (the "Software"),
* to deal in the Software without restriction, including without limitation
* the rights to use, copy, modify, merge, publish, distribute, sublicense,
* and/or sell copies of the Software, and to permit persons to whom the
* Software is furnished to do so, subject to the following conditions:
* *
* The above copyright notice and this permission notice shall be included in
* all copies or substantial portions of the Software.
* *
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
* THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT
* OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
* THE USE OR OTHER DEALINGS IN THE SOFTWARE.
* /

/**************************************************************************
* This file is based on code from bdf.c,v 1.22 2000/03/16 20:08:50
* *
* taken from Mark Leisher's xmbdfed package
* *
* /
--- FreeType 2 PSaux module is covered by the following notices ---


This software, and all works of authorship, whether in source or object code form as indicated by the copyright notice(s) included herein (collectively, the "Work") is made available, and may only be used, modified, and distributed under the FreeType Project License, LICENSE.TXT. Additionally, subject to the terms and conditions of the FreeType Project License, each contributor to the Work hereby grants to any individual or legal entity exercising permissions granted by the FreeType Project License and this section (hereafter, "You" or "Your") a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such contributor that are necessarily infringed by their contribution(s) alone or by combination of their contribution(s) with the Work to which such contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

By using, modifying, or distributing the Work you indicate that you have read and understood the terms and conditions of the FreeType Project License as well as those provided in this section, and you accept them fully.

---

### CUP Parser Generator License

Copyright 1996-2015 by Scott Hudson, Frank Flannery, C. Scott Ananian, Michael Petter

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both
the copyright notice and this permission notice and warranty disclaimer
appear in supporting documentation, and that the names of the authors or
their employers not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to
this software, including all implied warranties of merchantability and
fitness. In no event shall the authors or their employers be liable for
any special, indirect or consequential damages or any damages whatsoever
resulting from loss of use, data or profits, whether in an action of
contract, negligence or other tortious action, arising out of or in
connection with the use or performance of this software.

</pre>
This is the copyright file
## The Unicode Standard, Unicode Character Database, Version 13.0.0

### Unicode Character Database
```
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

See Terms of Use for definitions of Unicode Inc.'s
Data Files and Software.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S
DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),
YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE
TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE
THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2020 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

See Terms of Use for definitions of Unicode Inc.'s Data Files and Software.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2020 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or
(b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

jpackage test license file (just some sample text).

All images in this directory are copyright 1995 by Jeff Dinkins. Unauthorized reproduction is prohibited.

For more information about Jeff's photographs, please see:
  http://www.theFixx.org/Jeff

## Thai Dictionary

### Thai Dictionary License

<pre>
Copyright (C) 1982 The Royal Institute, Thai Royal Government.

Copyright (C) 1998 National Electronics and Computer Technology Center,
National Science and Technology Development Agency,
Ministry of Science Technology and Environment,
Thai Royal Government.

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

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

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

</pre>

## Apache Xerces v2.12.1

### Apache Xerces Notice

<pre>
===========================================================================
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
===========================================================================

Apache Xerces Java
Copyright 1999-2020 The Apache Software Foundation

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

Portions of this software were originally based on the following:
- voluntary contributions made by Paul Eng on behalf of the
  Apache Software Foundation that were originally developed at iClick, Inc.,
  software copyright (c) 1999.
</pre>

### Apache 2.0 License

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

</pre>

## Eastman Kodak Company: Portions of color management and imaging software

### Eastman Kodak Notice
Portions Copyright Eastman Kodak Company 1991-2003

### LCMS License

Little Color Management System
Copyright (c) 1998-2020 Marti Maria Saguer

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

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

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Files: *
Copyright: APPLICATION_COPYRIGHT
License: APPLICATION_LICENSE_TEXT
ADDITIONAL INFORMATION ABOUT LICENSING

Certain files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception.

Note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0 and may include FreeType. Such programs are licensed to you under their original licenses.
Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result.

Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Failing to distribute notices associated with some files may also create unexpected legal consequences.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

Copyright (c) %YEARS% Oracle and/or its affiliates. All rights reserved.
DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS FILE HEADER.

This code is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 2 only, as published by the Free Software Foundation. Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code.

This code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License version 2 for more details (a copy is included in the LICENSE file that accompanied this code).

You should have received a copy of the GNU General Public License version 2 along with this work; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.

Please contact Oracle, 500 Oracle Parkway, Redwood Shores, CA 94065 USA or visit www.oracle.com if you need additional information or have any questions.

## OASIS PKCS #11 Cryptographic Token Interface v2.40
Copyright (c) OASIS Open 2016. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [http://www.oasis-open.org/policies-guidelines/ipr]

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

---

## Apache Xalan v2.7.2

### Apache Xalan Notice

<pre>
==================================================================================
====
==  NOTICE file corresponding to the section 4d of the Apache License, Version 2.0, ==
==  in this case for the Apache Xalan distribution.                              ==
==================================================================================
====
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
</pre>
Specifically, we only include the XSLTC portion of the source from the Xalan distribution. The Xalan project has two processors: an interpretive one (Xalan Interpretive) and a compiled one (The XSLT Compiler (XSLTC)). We *only* use the XSLTC part of Xalan; We use the source from the packages that are part of the XSLTC sources.

Portions of this software was originally based on the following:

- voluntary contributions made by Ovidiu Predescu (ovidiu@cup.hp.com) on behalf of the Apache Software Foundation and was originally developed at Hewlett Packard Company.

### Apache 2.0 License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

JLEX COPYRIGHT NOTICE, LICENSE AND DISCLAIMER.
Copyright 1996-2003 by Elliot Joel Berk and C. Scott Ananian
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.
The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software. The portions of JLex output which are hard-coded into the JLex source code are (naturally)
Copyright (C) 1999-2007 Brian Paul   All Rights Reserved.

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

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

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

Attention, Contributors

When contributing to the Mesa project you must agree to the licensing terms of the component to which you're contributing.
The following section lists the primary components of the Mesa distribution and their respective licenses.
Mesa Component Licenses

<table>
<thead>
<tr>
<th>Component</th>
<th>Location</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Mesa code</td>
<td>src/mesa/</td>
<td>MIT</td>
</tr>
<tr>
<td>Device drivers</td>
<td>src/mesa/drivers/*</td>
<td>MIT, generally</td>
</tr>
<tr>
<td>Gallium code</td>
<td>src/gallium/</td>
<td>MIT</td>
</tr>
<tr>
<td>Ext headers</td>
<td>GL/glext.h</td>
<td>Khronos</td>
</tr>
<tr>
<td></td>
<td>GL/xlext.h</td>
<td>Khronos</td>
</tr>
<tr>
<td></td>
<td>GL/wglext.h</td>
<td>Khronos</td>
</tr>
</tbody>
</table>
----

include/GL/g1.h :

Mesa 3-D graphics library

Copyright (C) 1999-2006  Brian Paul  All Rights Reserved.
Copyright (C) 2009  VMware, Inc.  All Rights Reserved.

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

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

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

----

include/GL/g1ext.h
include/GL/g1xext.h
include/GL/wglxext.h :

Copyright (c) 2013 - 2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and/or associated documentation files (the
"Materials"), to deal in the Materials without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Materials, and to
permit persons to whom the Materials are furnished to do so, subject to
the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

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

*****************************************************************************

```

## PC/SC Lite v1.8.26

### PC/SC Lite License

Copyright (c) 1999-2003 David Corcoran <corcoran@linuxnet.com>
Copyright (c) 2001-2011 Ludovic Rousseau <ludovic.rousseau@free.fr>

```
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Changes to this license can be made only by the copyright author with explicit written consent.

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

</pre>

## jopt-simple v5.0.4

### MIT License

Copyright (c) 2004-2015 Paul R. Holser, Jr.

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

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

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

</pre>
## Cryptix AES v3.2.0

### Cryptix General License

Cryptix General License

Copyright (c) 1995-2005 The Cryptix Foundation Limited.
All rights reserved.

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

1. Redistributions of source code must retain the copyright notice, 
   this list of conditions and the following disclaimer.

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

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

</pre>
## JLine v3.14.0

### JLine License

JLine License


Copyright (c) 2002-2018, the original author or authors.
All rights reserved.

https://opensource.org/licenses/BSD-3-Clause

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

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

</pre>

## Apache Santuario v2.1.4

### Apache Santuario Notice

Apache Santuario - XML Security for Java
Copyright 1999-2019 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).
It was originally based on software copyright (c) 2001, Institute for Data Communications Systems, <http://www.nue.et-inf.uni-siegen.de/>.

The development of this software was partly funded by the European Commission in the <WebSig> project in the ISIS Programme.

### Apache 2.0 License

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"
The jQuery JavaScript Library v3.5.1 also includes Sizzle.js

Sizzle.js includes the following license:

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

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

The following license applies to all parts of this software except as documented below:

====

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

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

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

====

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

*******************

```
## DOM Level 3 Core Specification v1.0
```
W3C SOFTWARE NOTICE AND LICENSE

http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.

3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION. The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this
license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

</pre>

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE

===========================================

PNG Reference Library License version 2

* Copyright (c) 1995-2019 The PNG Reference Library Authors.
* Copyright (c) 2018-2019 Cosmin Truta.
* Copyright (c) 1996-1997 Andreas Dilger.
* Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.
PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

-----------------------------------------------

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are
Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are
derived from libpng-1.0.6, and are distributed according to the same
disclaimer and license as libpng-1.0.6 with the following individuals
added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabuddhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of
the library or against infringement. There is no warranty that our
efforts or the library will fulfill any of your particular purposes
or needs. This library is provided with all faults, and the entire
risk of satisfactory quality, performance, accuracy, and effort is
with the user.

Some files in the "contrib" directory and some configure-generated
files that are distributed with libpng have other copyright owners, and
are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are
Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from
libpng-0.96, and are distributed according to the same disclaimer and
license as libpng-0.96, with the following individuals added to the
list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88,
and are distributed according to the same disclaimer and license as
libpng-0.88, with the following individuals added to the list of
Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use
this source code in a product, acknowledgment is not required but would be appreciated.

All images in this directory are copyright 1995 by Jeff Dinkins.
## GIFLIB v5.2.1

```
### GIFLIB License
```

The GIFLIB distribution is Copyright (c) 1997 Eric S. Raymond

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

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

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

https://sourceforge.net/p/giflib/code/ci/master/tree/openbsd-reallocarray.c

Copyright (c) 2008 Otto Moerbeek <otto@drijf.net>
SPDX-License-Identifier: MIT
```

## libpng v1.6.37

### libpng License

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE
==============================================

PNG Reference Library License version 2
=========================================

* Copyright (c) 1995-2019 The PNG Reference Library Authors.
* Copyright (c) 2018-2019 Cosmin Truta.
* Copyright (c) 1996-1997 Andreas Dilger.
* Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)
-----------------------------------------------------------------------
libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabuddhe
Google Inc.
Vadim Barkov
and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

   Tom Lane
   Glenn Randers-Pehrson
   Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

   John Bowler
   Kevin Bracey
   Sam Bushell
   Magnus Holmgren
   Greg Roelofs
   Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

   Andreas Dilger
   Dave Martindale
   Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

</pre>

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or
translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a
work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR
INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERNED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be
called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

## Dynalink v.5

### Dynalink License

<pre>
Copyright (c) 2009-2013, Attila Szegedi

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the copyright holder nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

</pre>

### ASM License

<pre>
Copyright (c) 2000-2011 France Tlcom
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

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

</pre>

Copyright (c) %YEARS% Oracle and/or its affiliates. All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

- Neither the name of Oracle nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Copyright (c) %YEARS% Oracle and/or its affiliates. All rights reserved.

DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS FILE HEADER.

This code is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 2 only, as
This code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License version 2 for more details (a copy is included in the LICENSE file that accompanied this code).

You should have received a copy of the GNU General Public License version 2 along with this work; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.

Please contact Oracle, 500 Oracle Parkway, Redwood Shores, CA 94065 USA or visit www.oracle.com if you need additional information or have any questions.

### jQuery UI v1.12.1

#### jQuery UI License

Copyright jQuery Foundation and other contributors, https://jquery.org/

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

The following license applies to all parts of this software except as documented below:

====

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

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

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code contained within the demos directory.

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

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

---

## Independent JPEG Group: JPEG release 6b

### JPEG License

****************************************************************************
Copyright (C) 1991-1998, Thomas G. Lane.
This software is the work of Tom Lane, Philip Gladstone, Jim Boucher, Lee Crocker, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Guido Vollbeding, Ge’ Weijers, and other members of the Independent JPEG Group.

IJG is not affiliated with the official ISO JPEG standards committee.

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.) So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

********************************************************************
```
## International Components for Unicode (ICU4J) v67.1
```
COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2020 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

---------------------

Third-Party Software Licenses

This section contains third-party software notices and/or additional
terms for licensed third-party software components included within ICU
libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

From: https://www.unicode.org/copyright.html:

Unicode Copyright and Terms of Use

For the general privacy policy governing access to this site, see the Unicode Privacy Policy.

Unicode Copyright

Copyright 1991-2020 Unicode, Inc. All rights reserved.

Definitions

Unicode Data Files ("DATA FILES") include all data files under the directories:
https://www.unicode.org/Public/
https://www.unicode.org/reports/
https://www.unicode.org/ivd/data/
Unicode Data Files do not include PDF online code charts under the directory:
https://www.unicode.org/Public/

Unicode Software ("SOFTWARE") includes any source code published in the Unicode Standard
or any source code or compiled code under the directories:
https://www.unicode.org/Public/PROGRAMS/
https://www.unicode.org/Public/cldr/
http://site.icu-project.org/download/

Terms of Use

Certain documents and files on this website contain a legend indicating that "Modification is permitted." Any
person is hereby authorized, without fee, to modify such documents and files to create derivative works conforming
to the Unicode Standard, subject to Terms and Conditions herein.

Any person is hereby authorized, without fee, to view, use, reproduce, and distribute all documents and files,
subject to the Terms and Conditions herein.

Further specifications of rights and restrictions pertaining to the use of the Unicode DATA FILES and
SOFTWARE can be found in the Unicode Data Files and Software License.

Each version of the Unicode Standard has further specifications of rights and restrictions of use. For the book
editions (Unicode 5.0 and earlier), these are found on the back of the title page.

The Unicode PDF online code charts carry specific restrictions. Those restrictions are incorporated as the first
page of each PDF code chart.

All other files, including online documentation of the core specification for Unicode 6.0 and later, are
covered under these general Terms of Use.

No license is granted to "mirror" the Unicode website where a fee is charged for access to the "mirror" site.
Modification is not permitted with respect to this document. All copies of this document must be verbatim.

Restricted Rights Legend

Any technical data or software which is licensed to the United States of America, its agencies and/or
instrumentalities under this Agreement is commercial technical data or commercial computer software developed
exclusively at private expense as defined in FAR 2.101, or DFARS 252.227-7014 (June 1995), as applicable. For
technical data, use, duplication, or disclosure by the Government is subject to restrictions as set forth in DFARS
202.227-7015 Technical Data, Commercial and Items (Nov 1995) and this Agreement. For Software, in accordance
with FAR 12-212 or DFARS 227-7202, as applicable, use, duplication or disclosure by the Government is subject
to the restrictions set forth in this Agreement.

Warranties and Disclaimers

This publication and/or website may include technical or typographical errors or other inaccuracies. Changes
are periodically added to the information herein; these changes will be incorporated in new editions of the
publication and/or website. Unicode, Inc. may make improvements and/or changes in the product(s) and/or
program(s) described in this publication and/or website at any time.

If this file has been purchased on magnetic or optical media from Unicode, Inc. the sole and exclusive remedy
for any claim will be exchange of the defective media within ninety (90) days of original purchase.

EXCEPT AS PROVIDED IN SECTION E.2, THIS PUBLICATION AND/OR SOFTWARE IS PROVIDED
"AS IS" WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS, IMPLIED, OR STATUTORY,
INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR PURPOSE, OR NON-INFRINGEMENT. UNICODE, INC. AND ITS LICENSORS ASSUME NO
RESPONSIBILITY FOR ERRORS OR OMISSIONS IN THIS PUBLICATION AND/OR SOFTWARE OR
OTHER DOCUMENTS WHICH ARE REFERENCED BY OR LINKED TO THIS PUBLICATION OR THE
UNICODE WEBSITE.
Waiver of Damages

In no event shall Unicode, Inc. or its licensors be liable for any special, incidental, indirect or consequential damages of any kind, or any damages whatsoever, whether or not Unicode, Inc. was advised of the possibility of the damage, including, without limitation, those resulting from the following: loss of use, data or profits, in connection with the use, modification or distribution of this information or its derivatives.

Trademarks & Logos

The Unicode Word Mark and the Unicode Logo are trademarks of Unicode, Inc. The Unicode Consortium and Unicode, Inc. are trade names of Unicode, Inc. Use of the information and materials found on this website indicates your acknowledgement of Unicode, Inc.‘s exclusive worldwide rights in the Unicode Word Mark, the Unicode Logo, and the Unicode trade names.

The Unicode Consortium Name and Trademark Usage Policy (Trademark Policy) are incorporated herein by reference and you agree to abide by the provisions of the Trademark Policy, which may be changed from time to time in the sole discretion of Unicode, Inc.

All third party trademarks referenced herein are the property of their respective owners.

Miscellaneous

Jurisdiction and Venue. This website is operated from a location in the State of California, United States of America. Unicode, Inc. makes no representation that the materials are appropriate for use in other locations. If you access this website from other locations, you are responsible for compliance with local laws. This Agreement, all use of this website and any claims and damages resulting from use of this website are governed solely by the laws of the State of California without regard to any principles which would apply the laws of a different jurisdiction. The user agrees that any disputes regarding this website shall be resolved solely in the courts located in Santa Clara County, California. The user agrees said courts have personal jurisdiction and agree to waive any right to transfer the dispute to any other forum.

Modification by Unicode, Inc. Unicode, Inc. shall have the right to modify this Agreement at any time by posting it to this website. The user may not assign any part of this Agreement without Unicode, Inc.‘s prior written consent.

Taxes. The user agrees to pay any taxes arising from access to this website or use of the information herein, except for those based on Unicodes net income.

Severability. If any provision of this Agreement is declared invalid or unenforceable, the remaining provisions of this Agreement shall remain in effect.

Entire Agreement. This Agreement constitutes the entire agreement between the parties.

```
### Mozilla Public Suffix List

You are receiving a copy of the Mozilla Public Suffix List in the following file: `<java-home>/lib/security/public_suffix_list.dat`. The terms of the Oracle license do NOT apply to this file; it is licensed under the Mozilla Public License 2.0, separately from the Oracle programs you receive. If you do not wish to use the Public Suffix List, you may remove the `<java-home>/lib/security/public_suffix_list.dat` file.

The Source Code of this file is available under the Mozilla Public License, v. 2.0 and is located at https://raw.githubusercontent.com/publicsuffix/list/cbbba1d234670453df9c930dfbf510c0474d4301/public_suffix_list.dat.
```
If a copy of the MPL was not distributed with this file, you can obtain one at https://mozilla.org/MPL/2.0/.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

### MPL v2.0

Mozilla Public License Version 2.0

==================================

1. Definitions

1.1. "Contributor"
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
Software; or

(b) any new file in Source Code Form that contains any Covered
Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses
No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter
the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.
5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an    *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************
7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions
Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

-------------------------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

-------------------------------------------------------------

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
## Apache Commons Byte Code Engineering Library (BCEL) Version 6.4.1

### Apache Commons BCEL Notice
<pre>
=========================================================================
==  NOTICE file corresponding to the section 4 d of                    ==
==  the Apache License, Version 2.0,                                   ==
==  in this case for the Apache Commons BCEL distribution.             ==
=========================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
</pre>

### Apache 2.0 License
<pre>
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

</pre>

## Unicode Common Local Data Repository (CLDR) v37

### CLDR License

```
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

See Terms of Use for definitions of Unicode Inc.'s Data Files and Software.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE
```
THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2020 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

-- Terms of Use --

Unicode Copyright and Terms of Use
For the general privacy policy governing access to this site, see the Unicode Privacy Policy.

Unicode Copyright
Copyright 1991-2020 Unicode, Inc. All rights reserved.
Definitions
Unicode Data Files ("DATA FILES") include all data files under the directories:
https://www.unicode.org/Public/
Unicode Data Files do not include PDF online code charts under the directory:
https://www.unicode.org/Public/

Unicode Software ("SOFTWARE") includes any source code published in the Unicode Standard or any source code or compiled code under the directories:
https://www.unicode.org/Public/PROGRAMS/
https://www.unicode.org/Public/cldr/
http://site.icu-project.org/download/

Terms of Use
Certain documents and files on this website contain a legend indicating that "Modification is permitted." Any person is hereby authorized, without fee, to modify such documents and files to create derivative works conforming to the Unicode Standard, subject to Terms and Conditions herein.
Any person is hereby authorized, without fee, to view, use, reproduce, and distribute all documents and files, subject to the Terms and Conditions herein.
Further specifications of rights and restrictions pertaining to the use of the Unicode DATA FILES and SOFTWARE can be found in the Unicode Data Files and Software License.
Each version of the Unicode Standard has further specifications of rights and restrictions of use. For the book editions (Unicode 5.0 and earlier), these are found on the back of the title page.
The Unicode PDF online code charts carry specific restrictions. Those restrictions are incorporated as the first page of each PDF code chart.
All other files, including online documentation of the core specification for Unicode 6.0 and later, are covered under these general Terms of Use.
No license is granted to "mirror" the Unicode website where a fee is charged for access to the "mirror" site.
Modification is not permitted with respect to this document. All copies of this document must be verbatim.

Restricted Rights Legend
Any technical data or software which is licensed to the United States of America, its agencies and/or instrumentalities under this Agreement is commercial technical data or commercial computer software developed exclusively at private expense as defined in FAR 2.101, or DFARS 252.227-7014 (June 1995), as applicable. For technical data, use, duplication, or disclosure by the Government is subject to restrictions as set forth in DFARS 202.227-7015 Technical Data, Commercial and Items (Nov 1995) and this Agreement. For Software, in accordance with FAR 12-212 or DFARS 227-7202, as applicable, use, duplication or disclosure by the Government is subject to the restrictions set forth in this Agreement.

Warranties and Disclaimers
This publication and/or website may include technical or typographical errors or other inaccuracies. Changes are periodically added to the information herein; these changes will be incorporated in new editions of the publication and/or website. Unicode, Inc. may make improvements and/or changes in the product(s) and/or program(s) described in this publication and/or website at any time.
If this file has been purchased on magnetic or optical media from Unicode, Inc. the sole and exclusive remedy for any claim will be exchange of the defective media within ninety (90) days of original purchase.
EXCEPT AS PROVIDED IN SECTION E.2, THIS PUBLICATION AND/OR SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. UNICODE, INC. AND ITS LICENSORS ASSUME NO RESPONSIBILITY FOR ERRORS OR OMISSIONS IN THIS PUBLICATION AND/OR SOFTWARE OR OTHER DOCUMENTS WHICH ARE REFERENCED BY OR LINKED TO THIS PUBLICATION OR THE
UNICODE WEBSITE.

Waiver of Damages

In no event shall Unicode, Inc. or its licensors be liable for any special, incidental, indirect or consequential damages of any kind, or any damages whatsoever, whether or not Unicode, Inc. was advised of the possibility of the damage, including, without limitation, those resulting from the following: loss of use, data or profits, in connection with the use, modification or distribution of this information or its derivatives.

Trademarks & Logos

The Unicode Word Mark and the Unicode Logo are trademarks of Unicode, Inc. The Unicode Consortium and Unicode, Inc. are trade names of Unicode, Inc. Use of the information and materials found on this website indicates your acknowledgement of Unicode, Inc.'s exclusive worldwide rights in the Unicode Word Mark, the Unicode Logo, and the Unicode trade names.

The Unicode Consortium Name and Trademark Usage Policy (Trademark Policy) are incorporated herein by reference and you agree to abide by the provisions of the Trademark Policy, which may be changed from time to time in the sole discretion of Unicode, Inc.

All third party trademarks referenced herein are the property of their respective owners.

Miscellaneous

Jurisdiction and Venue. This website is operated from a location in the State of California, United States of America. Unicode, Inc. makes no representation that the materials are appropriate for use in other locations. If you access this website from other locations, you are responsible for compliance with local laws. This Agreement, all use of this website and any claims and damages resulting from use of this website are governed solely by the laws of the State of California without regard to any principles which would apply the laws of a different jurisdiction. The user agrees that any disputes regarding this website shall be resolved solely in the courts located in Santa Clara County, California. The user agrees said courts have personal jurisdiction and agree to waive any right to transfer the dispute to any other forum.

Modification by Unicode, Inc. Unicode, Inc. shall have the right to modify this Agreement at any time by posting it to this website. The user may not assign any part of this Agreement without Unicode, Inc.'s prior written consent.

Taxes. The user agrees to pay any taxes arising from access to this website or use of the information herein, except for those based on Unicodes net income.

Severability. If any provision of this Agreement is declared invalid or unenforceable, the remaining provisions of this Agreement shall remain in effect.

Entire Agreement. This Agreement constitutes the entire agreement between the parties.

```

1.4 log4j-api 2.17.0

1.4.1 Available under license:

Apache Log4j 1.x Compatibility API
Copyright 1999-2020 The Apache Software Foundation

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

Apache License
Version 2.0, January 2004

```
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.5 audit4j-spring 2.5.0
1.5.1 Available under license:

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.6 shadow 4.5-1ubuntu2.2

1.6.1 Available under license:

NOTE:
This license has been obsoleted by the change to the BSD-style copyright.
You may continue to use this license if you wish, but you are under no obligation to do so.

(*
This document is freely plagiarised from the 'Artistic Licence', distributed as part of the Perl v4.0 kit by Larry Wall, which is available from most major archive sites. I stole it from CrackLib.

$Id$
*)

This documents purpose is to state the conditions under which this Package (See definition below) viz: "Shadow", the Shadow Password Suite which is held by Julianne Frances Haugh, may be copied, such that the copyright holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.
So there.

***************************************************************************

Definitions:

A "Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification, or segments thereof.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when AND WHY you changed that file, and provided that you do at least ONE of the following:

a) place your modifications in the Public Domain or otherwise make them
Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

b) use the modified Package only within your corporation or organization.

c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide separate documentation for each non-standard executable that clearly documents how it differs from the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

b) accompany the distribution with the machine-readable source of the Package with your modifications.

c) accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. YOU MAY NOT CHARGE A FEE FOR THIS PACKAGE ITSELF. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that YOU DO NOT ADVERTISE this package as a product of your own.

6. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

7. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
The End

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

1.7 nettle 3.4.1-0ubuntu0.18.04.1
1.7.1 Available under license:
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run
modified versions of the software inside them, although the manufacturer
can do so. This is fundamentally incompatible with the aim of
protecting users' freedom to change the software. The systematic
pattern of such abuse occurs in the area of products for individuals to
use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those
products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions
of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.
States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to
avoid the special danger that patents applied to a free program could
make it effectively proprietary. To prevent this, the GPL assures that
patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and
modification follow.

TERMS AND CONDITIONS
0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of
interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
not control copyright. Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under
the conditions stated below. Sublicensing is not allowed; section 10
makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article
11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such
measures.

When you convey a covered work, you waive any legal power to forbid
circumvention of technological measures to the extent such circumvention
is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or
modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of
 technological measures.


You may convey verbatim copies of the Program's source code as you
receive it, in any medium, provided that you conspicuously and
appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any
non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to
produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified
it, and giving a relevant date.
b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical
medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.
Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.
You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered
work results from an entity transaction, each party to that
transaction who receives a copy of the work also receives whatever
licenses to the work the party’s predecessor in interest had or could
give under the previous paragraph, plus a right to possession of the
Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the
rights granted or affirmed under this License. For example, you may
not impose a license fee, royalty, or other charge for exercise of
rights granted under this License, and you may not initiate litigation
(including a cross-claim or counterclaim in a lawsuit) alleging that
any patent claim is infringed by making, using, selling, offering for
sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this
License of the Program or a work on which the Program is based. The
work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims
owned or controlled by the contributor, whether already acquired or
hereafter acquired, that would be infringed by some manner, permitted
by this License, of making, using, or selling its contributor version,
but do not include claims that would be infringed only as a
consequence of further modification of the contributor version. For
purposes of this definition, "control" includes the right to grant
patent sublicenses in a manner consistent with the requirements of
this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free
patent license under the contributor's essential patent claims, to
make, use, sell, offer for sale, import and otherwise run, modify and
propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express
agreement or commitment, however denominated, not to enforce a patent
(such as an express permission to practice a patent or covenant not to
sue for patent infringement). To "grant" such a patent license to a
party means to make such an agreement or commitment not to enforce a
patent against the party.

If you convey a covered work, knowingly relying on a patent license,
and the Corresponding Source of the work is not available for anyone
to copy, free of charge and under the terms of this License, through a
publicly available network server or other readily accessible means,
then you must either (1) cause the Corresponding Source to be so
available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.
13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```plaintext
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <https://www.gnu.org/philosophy/why-not-lgpl.html>.
```

### 1.8 spring-framework 4.3.22.RELEASE

#### 1.8.1 Available under license:

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

```plaintext
/*
 * Copyright 2002-2012 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```
Open Source Used In IOx-FogDirector-AC18 1.17.0

/*
* Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/serializer/Serializer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/io/ClassRelativeResourceLoader.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/util/comparator/NullSafeComparator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/SmartClassLoader.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/style/ToStringStyler.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/task/TaskRejectedException.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/convert/support/StringToArrayConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/util/PathMatcher.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/convert/support/CharacterToNumberFactory.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/convert/support/NumberToCharacterConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/classreading/MetadataReaderFactory.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/style/DefaultValueStyler.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/AttributeAccessor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/convert/support/CharacterToNumberFactory.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/convert/support/NumberToCharacterConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/serializable/Deserializable.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/core/ExceptionDepthComparator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
* jar/org/springframework/task/support/ConcurrentExecutorAdapter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
*/
jar/org/springframework/core/style/StylerUtils.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/task/TaskExecutor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/WritableResource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/DescriptiveResource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/FileSystemResourceLoader.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/xml/SimpleTransformErrorListener.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/comparator/ComparableComparator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/task/TaskTimeoutException.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/task/SyncTaskExecutor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/serializer/support/SerializationFailedException.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/comparator/BooleanComparator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/ErrorHandler.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/StringToBooleanConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/InfrastructureProxy.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/SpringVersion.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/ParameterNameDiscoverer.java

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

/*
 * Copyright 2002-2015 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/type/MethodMetadata.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/PriorityOrdered.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/support/FallbackObjectToStringConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/support/SerializationDelegate.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/InputStreamResource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/support/StringToCurrencyConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/StandardMethodMetadata.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/convert/converter/Converter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/DefaultParameterNameDiscoverer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/DefaultSerializer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/Ordered.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/convert/converter/Converter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/convert/DefaultParameterNameDiscoverer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/DefaultSerializer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/StandardMethodMetadata.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/serializer/support/DeserializingConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/serializer/support/SerializingConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/objenesis/SpringObjenesis.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/SystemEnvironmentPropertySource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/type/classreading/AnnotationMetadataReadingVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/serializer/DefaultDeserializer.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/ResizableByteArrayOutputStream.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/AlternativeJdkIdGenerator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/annotation/AnnotationConfigurationException.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/type/AnnotatedTypeMetadata.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/EnumerablePropertySource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/annotation/AliasFor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/annotation/SynthesizedAnnotation.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/StringToTimeZoneConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/StreamConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/framework/util/comparator/InstanceComparator.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/StringToLocaleConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/annotation/SynthesizedAnnotationInvocationHandler.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/framework/util/UpdateMessageDigestInputStream.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/converter/GenericConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/converter/ConditionalGenericConverter.java

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

/*
 * Copyright 2002-2019 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/ResolvableType.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/BridgeMethodResolver.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/convert/TypeDescriptor.java
No license file was found, but licenses were detected in source scan.

/**
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2011 INRIA, France Telecom
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the copyright holders nor the names of its
 *    contributors may be used to endorse or promote products derived from
 *    this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
 * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
 * THE POSSIBILITY OF SUCH DAMAGE.
*/
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Type.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/FieldWriter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Edge.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/FieldVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Handle.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ClassReader.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/AnnotationWriter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Handler.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ModuleWriter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Context.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Attribute.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ClassVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/CurrentFrame.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Label.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Opcodes.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ByteVector.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Frame.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/MethodWriter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/Item.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/AnnotationVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ClassWriter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/ModuleVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/asm/MethodVisitor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/PropertySourcesPropertyResolver.java

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

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/ReadOnlySystemAttributesMap.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/type/classreading/ClassMetadataReadingVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/InputStreamSource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/ConcurrencyThrottleSupport.java
Open Source Used In IOx-FogDirector-AC18 1.17.0  167

* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/concurrent/ListenableFutureTask.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/ClassPathResource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/ByteArrayResource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/LinkedCaseInsensitiveMap.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/StaxStreamXMLReader.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/MethodInvoker.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/MutablePropertySources.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/type/classreading/MethodMetadataReadingVisitor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/Resource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/Conventions.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/Env.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/MethodClassKey.java
/* Copyright 2002-2013 the original author or authors. */

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/type/classreading/SimpleMetadataReader.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/SystemPropertyUtils.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/StandardEnvironment.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/task/AsyncListenableTaskExecutor.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/task/AsyncTaskExecutor.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/xml/DomUtils.java

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

/*/ 
* Copyright 2002-2018 the original author or authors. 
* 
* Licensed under the Apache License, Version 2.0 (the "License"); 
* you may not use this file except in compliance with the License. 
* You may obtain a copy of the License at 
* http://www.apache.org/licenses/LICENSE-2.0 
*
* Unless required by applicable law or agreed to in writing, software 
* distributed under the License is distributed on an "AS IS" BASIS, 
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
* See the License for the specific language governing permissions and 
* limitations under the License. 
*/

Found in path(s):
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/MethodParameter.java
/.../PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/concurrent/ListenableFutureCallbackRegistry.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2010 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
jar/org/springframework/core/convert/ConversionException.java
*/opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
jar/org/springframework/core/io/support/VfsPatternUtils.java
No license file was found, but licenses were detected in source scan.

/***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2013 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
jar/org/springframework/asm/TypePath.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/asm/TypeReference.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/env/MapPropertySource.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/xml/XmlValidationModeDetector.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/ZonedDateTimeToCalendarConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/ArrayToCollectionConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/ArrayToObjectConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/SerializationUtils.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/lang/UsesJava8.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/support/ResourceArrayPropertyEditor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/io/support/ResourceArrayPropertyEditor.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/core/convert/support/ArrayToStringConverter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/lang/UsesJava7.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2.jar/org/springframework/util/backoff/BackOff.java
*/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/NestedIOException.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/util/Log4jConfigurer.java

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

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-jar/org/springframework/core/type/classreading/MetadataReader.java

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

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/env/SimpleCommandLineArgsParser.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/convert/support/ConvertingPropertyEditorAdapter.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/env/CommandLineArgs.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/env/PropertySources.java
* /opt/ws_local/PERMITS_SQL/1072735584_1595111289.37/0/spring-core-4-3-22-release-sources-2-
  jar/org/springframework/core/convert/support/ConfigurableConversionService.java

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

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/
* Copyright 2002-2008 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

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

/ *
* Copyright 2002-2007 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

1.9 findutils 4.6.0+git+20170828 2

1.9.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.
2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work’s users, your or third parties’ legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or
modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and
propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any
author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

1.10 jackson-databind 2.10.0
1.10.1 Available under license:

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

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

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.11 zjsonpatch 0.3.0

1.11.1 Available under license:

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

/*
 * Copyright 2016 flipkart.com zjsonpatch.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/NodeType.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/JsonDiff.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/NoopProcessor.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/ApplyProcessor.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/Constants.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/CompatibilityFlags.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/JsonPatchApplicationException.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/JsonPatchProcessor.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/InvalidJsonPatchException.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/Operation.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/JsonPatch.java
* /opt/cola/permits/1257333041_1642801938.9/0/zjsonpatch-0-3-0-sources-jar/com/flipkart/zjsonpatch/Diff.java

1.12 moshi 1.8.0

1.12.1 Available under license:

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

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

Copyright [yyyy] [name of copyright owner]

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

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

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.13 metrics-servlet 4.1.9
1.13.1 Available under license:
Apache-2.0

1.14 libtasn 4.13 2
1.14.1 Available under license:
@c The GNU Free Documentation License.
@center Version 1.3, 3 November 2008

@c This file is intended to be included within another document,
@c hence no sectioning command or @node.

@display
@uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
@end display

@enumerate 0
@item
PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document @dfn{free} in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or
whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

@item

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The ``Document'', below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as ``you''. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A ``Modified Version'' of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A ``Secondary Section'' is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The ``Invariant Sections'' are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The ``Cover Texts'' are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A ``Transparent'' copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of
pixels) generic paint programs or (for drawings) some widely available
drawing editor, and that is suitable for input to text formatters or
for automatic translation to a variety of formats suitable for input
to text formatters. A copy made in an otherwise Transparent file
format whose markup, or absence of markup, has been arranged to thwart
or discourage subsequent modification by readers is not Transparent.
An image format is not Transparent if used for any substantial amount
of text. A copy that is not ``Transparent'' is called ``Opaque''.

Examples of suitable formats for Transparent copies include plain
ASCII without markup, Texinfo input format, \LaTeX{} input
format, SGML or XML using a publicly available
DTD, and standard-conforming simple HTML,
PostScript or PDF designed for human modification. Examples
of transparent image formats include PNG, XCF and
JPG@. Opaque formats include proprietary formats that can be
read and edited only by proprietary word processors, SGML or
XML for which the DTD and/or processing tools are
not generally available, and the machine-generated HTML,
PostScript or PDF produced by some word processors for
output purposes only.

The ```Title Page'' means, for a printed book, the title page itself,
plus such following pages as are needed to hold, legibly, the material
this License requires to appear in the title page. For works in
formats which do not have any title page as such, ```Title Page'' means
the text near the most prominent appearance of the work's title,
preceding the beginning of the body of the text.

The ```publisher'' means any person or entity that distributes copies
of the Document to the public.

A section ```Entitled XYZ'' means a named subunit of the Document whose
title either is precisely XYZ or contains XYZ in parentheses following
text that translates XYZ in another language. (Here XYZ stands for a
specific section name mentioned below, such as ```Acknowledgements'',
```Dedications'', ```Endorsements'', or ```History''.) To ```Preserve the Title''
of such a section when you modify the Document means that it remains a
section ```Entitled XYZ'' according to this definition.

The Document may include Warranty Disclaimers next to the notice which
states that this License applies to the Document. These Warranty
Disclaimers are considered to be included by reference in this
License, but only as regards disclaiming warranties: any other
implication that these Warranty Disclaimers may have is void and has
no effect on the meaning of this License.

@item
VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.
It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

- State on the Title page the name of the publisher of the Modified Version, as the publisher.

- Preserve all the copyright notices of the Document.

- Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

- Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

- Preserve in that license notice the full lists of Invariant Sections
and required Cover Texts given in the Document's license notice.

@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled ``History'', Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled ``History'' in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

@item
For any section Entitled ``Acknowledgements'' or ``Dedications'', Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

@item
Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

@item
Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

@item
Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

@item
Preserve any Warranty Disclaimers.
@end enumerate

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material
copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled ``Endorsements'', provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

@item

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ``History'' in the various original documents, forming one section Entitled ``History''; likewise combine any sections Entitled ``Acknowledgements'', and any sections Entitled ``Dedications''. You must delete all
sections Entitled "Endorsements."

@item
COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

@item
AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

@item
TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions
of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

@item
TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

@item
FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this
License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

@item

RE LICENSING

``Massive Multiauthor Collaboration Site'' (or ``MMC Site'') means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A ``Massive Multiauthor Collaboration'' (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

``CC-BY-SA'' means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

``Incorporate'' means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is ``eligible for relicensing'' if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

@end enumerate

@page
@heading ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and
license notices just after the title page:

@smallexample
@group
Copyright (C) @var{year} @var{your name}.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled ``GNU Free Documentation License''.
@end group
@end smallexample

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the ``with@dots{}Texts.''@: line with this:

@smallexample
@group
with the Invariant Sections being @var{list their titles}, with
the Front-Cover Texts being @var{list}, and with the Back-Cover Texts
being @var{list}.
@end group
@end smallexample

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

@c Local Variables:
@c ispell-local-pdict: "ispell-dict"
@c End:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages—typically libraries—of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an
exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.
The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no
You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in
reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.
Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.
11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license
you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.
If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.15 nspr 4.18 1ubuntu1
1.15.1 Available under license:
Mozilla Public License Version 2.0
==================================
1. Definitions
--------------
1.1. "Contributor"
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means
(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
(b) that the Covered Software was made available under the terms of
version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of
This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code
Form, as described in Section 3.1, and You must inform recipients of
the Executable Form how they can obtain a copy of such Source Code
Form by reasonable means in a timely manner, at a charge no more
than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this
License, or sublicense it under different terms, provided that the
license for the Executable Form does not attempt to limit or alter
the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice,
provided that You also comply with the requirements of this License for
the Covered Software. If the Larger Work is a combination of Covered
Software with a work governed by one or more Secondary Licenses, and the
Covered Software is not Incompatible With Secondary Licenses, this
License permits You to additionally distribute such Covered Software
under the terms of such Secondary License(s), so that the recipient of
the Larger Work may, at their option, further distribute the Covered
Software under the terms of either this License or such Secondary
License(s).

3.4. Notices

You may not remove or alter the substance of any license notices
(including copyright notices, patent notices, disclaimers of warranty,
or limitations of liability) contained within the Source Code Form of
the Covered Software, except that You may alter any license notices to
the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support,
indemnity or liability obligations to one or more recipients of Covered
Software. However, You may do so only on Your own behalf, and not on
behalf of any Contributor. You must make it absolutely clear that any
such warranty, support, indemnity, or liability obligation is offered by
You alone, and You hereby agree to indemnify every Contributor for any
liability incurred by such Contributor as a result of warranty, support,
indemnity or liability terms You offer. You may include additional
disclaimers of warranty and limitations of liability specific to any
jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this
License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"      *
*  basis, without warranty of any kind, either expressed, implied, or *
*  statutory, including, without limitation, warranties that the      *
*  Covered Software is free of defects, merchantable, fit for a       *
*  particular purpose or non-infringing. The entire risk as to the    *
*                                                                      *
************************************************************************
* quality and performance of the Covered Software is with You. * 
* Should any Covered Software prove defective in any respect, You * 
* (not any Contributor) assume the cost of any necessary servicing, * 
* repair, or correction. This disclaimer of warranty constitutes an * 
* essential part of this License. No use of any Covered Software is * 
* authorized under this License except under this disclaimer. * 
* *
* **************************************************************************
* **************************************************************************
* 7. Limitation of Liability
* --------------------------
* *
* Under no circumstances and under no legal theory, whether tort * 
* (including negligence), contract, or otherwise, shall any * 
* Contributor, or anyone who distributes Covered Software as * 
* permitted above, be liable to You for any direct, indirect, * 
* special, incidental, or consequential damages of any character * 
* including, without limitation, damages for lost profits, loss of * 
* goodwill, work stoppage, computer failure or malfunction, or any * 
* and all other commercial damages or losses, even if such party * 
* shall have been informed of the possibility of such damages. This * 
* limitation of liability shall not apply to liability for death or * 
* personal injury resulting from such party's negligence to the * 
* extent applicable law prohibits such limitation. Some * 
* jurisdictions do not allow the exclusion or limitation of * 
* incidental or consequential damages, so this exclusion and * 
* limitation may not apply to You. *
* *
* **************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous
------------------

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides 
that the language of a contract shall be construed against the drafter 
shall not be used to construe this License against a Contributor.

10. Versions of the License

-----------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 
10.3, no one other than the license steward has the right to modify or 
publish new versions of this License. Each version will be given a 
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version 
of the License under which You originally received the Covered Software, 
or under the terms of any subsequent version published by the license 
steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to 
create a new license for such software, you may create and use a 
modified version of this License if you rename the license and remove 
any references to the name of the license steward (except to note that 
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary 
Licenses

If You choose to distribute Source Code Form that is Incompatible With 
Secondary Licenses under the terms of this version of the License, the 
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

-------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public 
License, v. 2.0. If a copy of the MPL was not distributed with this 
file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular 
file, then You may include the notice in a location (such as a LICENSE 
file in a relevant directory) where a recipient would be likely to look 
for such a notice.
You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

1.16 commons-logging 1.1.3

1.16.1 Available under license:
Apache Commons Logging
Copyright 2003-2013 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.17 kubernetes-model-flowcontrol 5.4.1

1.17.1 Available under license:

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

Copyright (C) 2015 Red Hat, Inc.

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

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2015 Red Hat, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License”);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/LimitedPriorityLevelConfiguration.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/LimitResponse.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/ServiceAccountSubject.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfigurationCondition.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfigurationStatus.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/UserSubject.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfiguration.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/GroupSubject.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/Subject.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfigurationReference.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/QueuingConfiguration.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfigurationSpec.java
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/flowcontrol/v1beta1/PriorityLevelConfigurationList.java

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

* Copyright (C) 2015 Red Hat, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* 

Found in path(s):
* /opt/cola/permits/1283430890_1646722529.57/0/kubernetes-model-flowcontrol-5-4-1-sources-jar/manifest.vm
1.18 pcre 8.39 9

1.18.1 Available under license:

PCRE LICENCE

---------

PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself. The data in the testdata directory is not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions, and a just-in-time compiler that can be used to optimize pattern matching. These are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS

-----------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2019 University of Cambridge
All rights reserved.

PCRE JUST-IN-TIME COMPILATION SUPPORT

-------------------------------------

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2010-2019 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER
THE C++ WRAPPER FUNCTIONS

Contributed by: Google Inc.

Copyright (c) 2007-2012, Google Inc.
All rights reserved.

THE "BSD" LICENCE

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

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

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

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

1.19 joda-time 2.8.2

1.19.1 Available under license:

==============================================================================
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
==============================================================================

This product includes software developed by Joda.org (http://www.joda.org/).

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.20 file 5.32-2ubuntu0.4
1.20.1 Available under license:

$File: COPYING,v 1.1 2008/02/05 19:08:11 christos Exp$
Software written by Ian F. Darwin and others;
maintained 1994- Christos Zoulas.

This software is not subject to any export provision of the United States
Department of Commerce, and may be exported to any country or planet.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice immediately at the beginning of the file, without modification,
   this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

1.21 metrics-jvm 4.1.9
1.21.1 Available under license:
Apache-2.0

1.22 commons-lang3 3.5
1.22.1 Available under license:

Apache Commons Lang
Copyright 2001-2016 The Apache Software Foundation

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

This product includes software from the Spring Framework,
under the Apache License 2.0 (see: StringUtils.containsWhitespace())

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.23 httpcore5 5.0
1.23.1 Available under license :

Apache HttpComponents Core HTTP/1.1
Copyright 2005-2020 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.24 javax-persistence-api 2.2

1.24.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008, 2009 Sun Microsystems. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 The Eclipse Public License is available at http://www.eclipse.org/legal/epl-
Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2011 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008, 2010 Oracle. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2015 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2017 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html and the Eclipse Distribution License is available at'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008 - 2013 Oracle Corporation. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v1.0 and Eclipse Distribution License v. 1.0 * The Eclipse Public License is available at http://www.eclipse.org/legal/epl-v10.html * and the Eclipse Distribution License is available at'
1.25 qemu-utils 2.11+dfsg-1ubuntu7.39

1.25.1 Available under license:

/*
 * FreeBSD strace list
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, see <http://www.gnu.org/licenses/>.
 */

{ TARGET_FREEBSD_NR___acl_aclcheck_fd, "__acl_aclcheck_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_aclcheck_file, "__acl_aclcheck_file", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_aclcheck_link, "__acl_aclcheck_link", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_fd, "__acl_delete_fd", "%s(%d, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_file, "__acl_delete_file", "%s("\"%s\", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_link, "__acl_delete_link", "%s("\"%s\", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_fd, "__acl_get_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_file, "__acl_get_file", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_link, "__acl_get_link", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_fd, "__acl_set_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_file, "__acl_set_file", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_link, "__acl_set_link", "%s("\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___semctl, "__semctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___sysctl, "__sysctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___umtx_op, "__umtx_op", "%s(%#x, %d, %d, %#x, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_accept, "accept", "%s(%d,%#x,%#x)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_access, "access", "%s("%s",%#o)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_acct, "acct", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_adjtime, "adjtime", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_bind, "bind", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_break, "break", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_chdir, "chdir", "%s("%s")" }, NULL, NULL 
{ TARGET_FREEBSD_NR_chflags, "chflags", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_chmod, "chmod", "%s("%s",%#o)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_chown, "chown", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_chroot, "chroot", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_clock_getres, "clock_getres", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_clock_gettime, "clock_gettime", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_clock_settime, "clock_settime", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_close, "close", "%s(%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_connect, "connect", "%s(%d,%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_dup, "dup", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_dup2, "dup2", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_eaccess, "eaccess", "%s("%s",%#x)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_execve, "execve", NULL, print_execve, NULL 
{ TARGET_FREEBSD_NR_exit, "exit", "%s(%d)\n" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattrctl, "extattrctl", "%s("%s",%d,\"%s\",%d,\"%s\")", NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_delete_fd, "extattr_delete_fd", "%s(%d,%d,%\"%s\")" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_delete_file, "extattr_delete_file", "%s("%s",%d,%\"%s\")" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_delete_link, "extattr_delete_link", "%s("%s",%d,%\"%s\")" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_get_fd, "extattr_get_fd", "%s(%d,%d,%\"%s\")" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_get_file, "extattr_get_file", "%s("%s",%d,%\"%s\",%d,%\"%s\",%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_get_link, "%s("%s",%d,%\"%s\",%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_list_fd, "extattr_list_fd", "%s(%d,%d,%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_list_file, "extattr_list_file", "%s("%s",%d,%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_list_link, "extattr_list_link", "%s("%s",%d,%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_set_fd, "extattr_set_fd", "%s(%d,%d,%\"%s\",%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_set_file, "extattr_set_file", "%s("%s",%d,%\"%s\",%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_extattr_set_link, "%s("%s",%d,%\"%s\",%#x,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_fchdir, "fchdir", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_fchflags, "fchflags", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_fchmod, "fchmod", "%s(%d,%#o)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_fchown, "fchown", "%s(%d,%d,%d)" }, NULL, NULL 
{ TARGET_FREEBSD_NR_fcntl, "fcntl", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_fexecve, "fexecve", NULL, print_execve, NULL 
{ TARGET_FREEBSD_NR_fopen, "fopen", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_fstat, "fstat", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_fstatfs, "fstatfs", NULL, NULL, NULL 
{ TARGET_FREEBSD_NR_flock, "flock", NULL, NULL, NULL 

---
[ TARGET_FREEBSD_NR_fork, "fork", "%s()", NULL, NULL ],
[ TARGET_FREEBSD_NR_fpathconf, "fpathconf", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_fstat, "fstat", "%s(%d,%#x)", NULL, NULL ],
[ TARGET_FREEBSD_NR_fstatat, "fstatat", "%s(%d,"%s", %#x, %d)", NULL, NULL ],
[ TARGET_FREEBSD_NR_fstatfs, "fstatfs", "%s(%d,%#x)", NULL, NULL ],
[ TARGET_FREEBSD_NR_fsync, "fsync", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_ftruncate, "ftruncate", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_futimes, "futimes", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getcontext, "getcontext", "%s(%#x)", NULL, NULL ],
[ TARGET_FREEBSD_NR_getdirentries, "getdirentries", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_freebsd6_mmap, "freebsd6_mmap", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getegid, "getegid", "%s()", NULL, NULL ],
[ TARGET_FREEBSD_NR_geteuid, "geteuid", "%s()", NULL, NULL ],
[ TARGET_FREEBSD_NR_getfh, "getfh", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getfsstat, "getfsstat", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getgid, "getgid", "%s()", NULL, NULL ],
[ TARGET_FREEBSD_NR_getgroups, "getgroups", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getitimer, "getitimer", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getlogin, "getlogin", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getpeername, "getpeername", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getpgid, "getpgid", "getpgid", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getpriority, "getpriority", "%s(%#x,%#x)", NULL, NULL ],
[ TARGET_FREEBSD_NR_getresgid, "getresgid", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getresuid, "getresuid", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getrlimit, "getrlimit", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getrusage, "getrusage", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getsid, "getsid", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getsockname, "getsockname", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_getsockopt, "getsockopt", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_lchown, "lchown", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_link, "link", "%s(""%s","%s")", NULL, NULL ],
[ TARGET_FREEBSD_NR_lstat, "lstat", "%s(""%s",%p)", NULL, NULL ],
[ TARGET_FREEBSD_NR_mincore, "mincore", "mincore", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_listen, "listen", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_ioctl, "ioctl", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_lseek, "lseek", NULL, NULL, NULL ],
[ TARGET_FREEBSD_NR_madvise, "madvise", NULL, NULL, NULL ]
{ TARGET_FREEBSD_NR_minherit, "minherit", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mkdir, "mkdir", "%s(\"%s\",%#o)\", NULL, NULL },
{ TARGET_FREEBSD_NR_mkfifo, "mkfifo", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mknod, "mknod", "%s(\"%s\",%#o,%#x)\", NULL, NULL },
{ TARGET_FREEBSD_NR_mlock, "mlock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mlockall, "mlockall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mmap, "mmap", NULL, NULL, print_syscall_ret_addr },
{ TARGET_FREEBSD_NR_mount, "mount", "mount", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mprotect, "mprotect", "%s(%#x,%#x,%d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_msgctl, "msgctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msgget, "msgget", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msgrcv, "msgrcv", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mgsnd, "msgsnd", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msync, "msync", "msync", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munlock, "munlock", "munlock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munlockall, "munlockall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munmap, "munmap", "%s(%p,%d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_nanosec, "nanosec", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_nfssvc, "nfssvc", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_open, "open", "%s(\"%s\",%#x,%#o)\", NULL, NULL },
{ TARGET_FREEBSD_NR_openat, "openat", "%s(%d, \"%s\",%#x,%#o)\", NULL, NULL },
{ TARGET_FREEBSD_NR_pathconf, "pathconf", "%s(\"%s\",%#d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_pipe, "pipe", "pipe", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_poll, "poll", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pread, "pread", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_preadv, "preadv", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_prof, "prof", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ptrace, "ptrace", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwrite, "pwrite", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwritev, "pwritev", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_quotactl, "quotactl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_read, "read", "%s(%d,%#x,%d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_readlink, "readlink", "%s(\"%s\",%#p,%#d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_readv, "readv", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_reboot, "reboot", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_recvfrom, "recvfrom", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_recvmsg, "recvmsg", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rename, "rename", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_revoke, "revoke", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rfork, "rfork", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rmdir, "rmdir", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rtprio_thread, "rtprio_thread", "%s(%d,%d,%p)\", NULL, NULL },
{ TARGET_FREEBSD_NR_sbrk, "sbrk", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sched_yield, "sched_yield", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_select, "select", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_semget, "semget", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_semop, "semop", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sendmsg, "sendmsg", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sendto, "sendto", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setcontext, "setcontext", "%s(%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_setegid, "setegid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_seteuid, "seteuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setgid, "setgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setgroups, "setgroups", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setitimer, "setitimer", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setlogin, "setlogin", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setpgid, "setpgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setpriority, "setpriority", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setreuid, "setreuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setregid, "setregid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setresgid, "setresgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setresuid, "setresuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setsid, "setsid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setsockopt, "setsockopt", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_settimeofday, "settimeofday", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setuid, "setuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_shmat, "shmat", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_shmctl, "shmctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_shmdt, "shmctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_shmget, "shmget", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_socket, "socket", "%s(%d,%d,%d)", NULL, NULL },
{ TARGET_FREEBSD_NR_socketpair, "socketpair", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sstk, "sstk", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sqrt, "sqrt", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_stat, "stat", "%s("%s",%p)", NULL, NULL },
{ TARGET_FREEBSD_NR_statfs, "statfs", "%s("%s",%p)", NULL, NULL },
{ TARGET_FREEBSD_NR_symlink, "symlink", "%s("%s","%s")", NULL, NULL },
{ TARGET_FREEBSD_NR_sync, "sync", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sysarch, "sysarch", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_syscall, "syscall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_thr_create, "thr_create", "%s(%#x, %#x, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_exit, "thr_exit", "%s(%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_kill, "thr_kill", "%s(%d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_kill2, "thr_kill2", "%s(%d, %d, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_new, "thr_new", "%s(%#x, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_self, "thr_self", "%s(%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_set_name, "thr_set_name", "%s(%d, %s)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_suspend, "thr_suspend", "%s(%d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_thr_wake, "thr_wake", "%s(%d)", NULL, NULL },
NOTE! This copyright does *not* cover user programs that use kernel services by normal system calls - this is merely considered normal use of the kernel, and does *not* fall under the heading of "derived work". Also note that the GPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the Linux kernel) is copyrighted by me and others who actually wrote it.

Also note that the only valid version of the GPL as far as the kernel is concerned is _this_ particular version of the license (ie v2, not v2.2 or v3.x or whatever), unless explicitly otherwise stated.

Linus Torvalds

----------------------------------------

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OperATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
Slirp was written by Danny Gasparovski.
Copyright (c), 1995,1996 All Rights Reserved.

Slirp is maintained by Kelly Price <tygris+slirp@erols.com>

Slirp is free software; "free" as in you don't have to pay for it, and you are free to do whatever you want with it. I do not accept any donations, monetary or otherwise, for Slirp. Instead, I would ask you to pass this potential donation to your favorite charity. In fact, I encourage *everyone* who finds Slirp useful to make a small donation to their favorite charity (for example, GreenPeace). This is not a requirement, but a suggestion from someone who highly values the service they provide.

The copyright terms and conditions:

---BEGIN---

Copyright (c) 1995,1996 Danny Gasparovski. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

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

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

---END---

This basically means you can do anything you want with the software, except 1) call it your own, and 2) claim warranty on it. There is no warranty for this software. None. Nada. If you lose a million dollars while using Slirp, that's your loss not mine. So, ***USE AT YOUR OWN RISK!***.

If these conditions cannot be met due to legal restrictions (E.g. where it is against the law to give out Software without warranty), you must cease using the software and delete all copies you have.

Slirp uses code that is copyrighted by the following people/organizations:

Juha Pirkola.
Gregory M. Christy.
The Regents of the University of California.
Carnegie Mellon University.
The Australian National University.
RSA Data Security, Inc.

Please read the top of each source file for the details on the various copyrights.
Copyright (c) Individual contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PyCA Cryptography nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
      stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
      whole or in part contains or is derived from the Program or any
      part thereof, to be licensed as a whole at no charge to all third
      parties under the terms of this License.

   c) If the modified program normally reads commands interactively
      when run, you must cause it, when started running for such
      interactive use in the most ordinary way, to print or display an
      announcement including an appropriate copyright notice and a
      notice that there is no warranty (or else, saying that you provide
      a warranty) and that users may redistribute the program under
      these conditions, and telling the user how to view a copy of this
      License.  (Exception: if the Program itself is interactive but
      does not normally print such an announcement, your work based on
      the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole.  If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works.  But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

The following points clarify the QEMU license:

1) QEMU as a whole is released under the GNU General Public License,
   version 2.

2) Parts of QEMU have specific licenses which are compatible with the
   GNU General Public License, version 2. Hence each source file contains
   its own licensing information. Source files with no licensing information
   are released under the GNU General Public License, version 2 or (at your
   option) any later version.

   As of July 2013, contributions under version 2 of the GNU General Public
   License (and no later version) are only accepted for the following files
   or directories: bsd-user/, linux-user/, hw/vfio/, hw/xen/xen_pt*.

3) The Tiny Code Generator (TCG) is released under the BSD license
   (see license headers in files).

4) QEMU is a trademark of Fabrice Bellard.

Fabrice Bellard and the QEMU team

A. HISTORY OF THE SOFTWARE
   =============================

Open Source Used In IOx-FogDirector-AC18 1.17.0 290
Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see http://www.cnri.reston.va.us) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations (now Zope Corporation, see http://www.zope.com). In 2001, the Python Software Foundation (PSF, see http://www.python.org/psf/) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation is a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived from</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>1.3 thru 1.5.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2</td>
<td>2.1.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.1</td>
<td>2.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.2</td>
<td>2.2.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.3</td>
<td>2.2.2</td>
<td>2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3</td>
<td>2.2.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.1</td>
<td>2.3</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.2</td>
<td>2.3.1</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.3</td>
<td>2.3.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.4</td>
<td>2.3.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.5</td>
<td>2.3.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4</td>
<td>2.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>
2.4.1  2.4  2005  PSF  yes
2.4.2  2.4.1  2005  PSF  yes
2.4.3  2.4.2  2006  PSF  yes
2.5  2.4  2006  PSF  yes
2.7  2.6  2010  PSF  yes

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
-------------------------------------------
BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS"
basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
---------------------------------------

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 (“CNRI”), and the Individual or Organization (“Licensee”) accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License
3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be
bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILUR OR THER LIBRARY TO OPORATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBLITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
That's all there is to it!

LICENCE
========

The software in this repository is covered by the following licence.

// Copyright 2013, ARM Limited
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice,
//   this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice,
//   this list of conditions and the following disclaimer in the documentation
//   and/or other materials provided with the distribution.
// * Neither the name of ARM Limited nor the names of its contributors may be
//   used to endorse or promote products derived from this software without
//   specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS CONTRIBUTORS "AS IS" AND
// ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
// WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
// DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
// FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
// DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
// SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
// CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
// OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

argparse is (c) 2006-2009 Steven J. Bethard <steven.bethard@gmail.com>.

The argparse module was contributed to Python as of Python 2.7 and thus was licensed under the Python license. Same license applies to all files in the argparse package project.

For details about the Python License, please see doc/Python-License.txt.

History
-------

Before (and including) argparse 1.1, the argparse package was licensed under Apache License v2.0.

After argparse 1.1, all project files from the argparse project were deleted due to license compatibility issues between Apache License 2.0 and GNU GPL v2.
The project repository then had a clean start with some files taken from Python 2.7.1, so definitely all files are under Python License now. All the files in this directory and subdirectories are released under a BSD like license (see header in each file). No other license is accepted.

1.26 listenablefuture 9999.0-empty-to-avoid-conflict-with-guava
1.26.1 Available under license:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

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

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

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

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

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

END OF TERMS AND CONDITIONS

1.27 libasnheimdal 7.5.0+dfsg-1

1.27.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

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

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in /usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
# .
# 1. The authors are not responsible for the consequences of use of this
#   software, no matter how awful, even if they arise from flaws in it.
# .
# 2. The origin of this software must not be misrepresented, either by
#   explicit claim or by omission. Since few users ever read sources,
#   credits must appear in the documentation.
# .
# 3. Altered versions must be plainly marked as such, and must not be
#   misrepresented as being the original software. Since few users
#   ever read sources, credits must appear in the documentation.
# .
# 4. This notice may not be removed or altered.

@macro copynext{} 
@vskip 20pt plus 1fil 
@end macro

@macro copyrightstart{} 
@end macro

@macro copyrightend{} 
@end macro

@node Copyrights and Licenses, , Acknowledgments, Top
@comment node-name, next, previous, up
@appendix Copyrights and Licenses

@heading Kungliga Tekniska Hgskolan

@copyrightstart 
@verbatim
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).  
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

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

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

@end verbatim 
@copyrightend

Open Source Used In IOx-FogDirector-AC18 1.17.0 314
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@copynext

@heading Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

@verbatim

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

@end verbatim
The parts of the libroken, most of libtelnet, telnet, ftp, and popper.

Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

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

@end verbatim
@end copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim
@end copynext

@heading Doug Rabson
GSS-API mechglue layer.

@verbatim

Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@end verbatim

@heading PADL Software Pty Ltd

@verbatim

@item GSS-API CFX, SPNEGO, naming extensions, API extensions.
@item KCM credential cache.
@item HDB LDAP backend.
@end verbatim

Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim

Copyright (c) 2005 Marko Kreen
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim
Copyright (c) 2006,2007 NTT (Nippon Telegraph and Telephone Corporation) . All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@heading The NetBSD Foundation, Inc.

vis.c in libbroken

@verbatim
Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim
rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>
This code is hereby placed in the public domain.

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

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim

Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim
@copynext
@heading Richard Outerbridge

DES core in libhcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc.
All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

@end verbatim

@copynext

@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
1.28 fdisk 2.31.1
1.28.1 Available under license:

/*
 * Copyright (c) 1989 The Regents of the University of California.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *    This product includes software developed by the University of
 *    California, Berkeley and its contributors.
 * 4. Neither the name of the University nor the names of its contributors
 *    may be used to endorse or promote products derived from this software
 *    without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 */

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

The complete text of the license is available in the
../Documentation/licenses/COPYING.LGPLv2.1 file.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and 
distribute such modifications or work under the terms of Section 1 
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices 
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in 
whole or in part contains or is derived from the Program or any 
part thereof, to be licensed as a whole at no charge to all third 
parties under the terms of this License.

c) If the modified program normally reads commands interactively 
when run, you must cause it, when started running for such 
interactive use in the most ordinary way, to print or display an 
announcement including an appropriate copyright notice and a 
otice that there is no warranty (or else, saying that you provide 
a warranty) and that users may redistribute the program under 
these conditions, and telling the user how to view a copy of this 
License. (Exception: if the Program itself is interactive but 
does not normally print such an announcement, your work based on 
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If 
identifiable sections of that work are not derived from the Program, 
and can be reasonably considered independent and separate works in 
themselves, then this License, and its terms, do not apply to those 
sections when you distribute them as separate works. But when you 
distribute the same sections as part of a whole which is a work based 
on the Program, the distribution of the whole must be on the terms of 
this License, whose permissions for other licensees extend to the 
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest 
your rights to work written entirely by you; rather, the intent is to 
exercise the right to control the distribution of derivative or 
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program 
with the Program (or with a work based on the Program) on a volume of 
a storage or distribution medium does not bring the other work under 
the scope of this License.

3. You may copy and distribute the Program (or a work based on it, 
under Section 2) in object code or executable form under the terms of 
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

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

1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This library is free software; you can redistribute it and/or
modify it under the terms of the Modified BSD License.

The complete text of the license is available in the
../Documentation/licenses/COPYING.BSD-3 file.

1.29 gmp 6.1.2+dfsg-2

1.29.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.
"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that
Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.
Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this
License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.
You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that
country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation
Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices"
to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by the
Corresponding Source fixed on a durable physical medium
customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by a
written offer, valid for at least three years and valid for as
long as you offer spare parts or customer support for that product
model, to give anyone who possesses the object code either (1) a
copy of the Corresponding Source for all the software in the
product that is covered by this License, on a durable physical
medium customarily used for software interchange, for a price no
more than your reasonable cost of physically performing this
conveying of source, or (2) access to copy the
Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the
written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord
with subsection 6b.

d) Convey the object code by offering access from a designated
place (gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party)
that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the
Corresponding Source. Regardless of what server hosts the
Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided
you inform other peers where the object code and Corresponding
Source of the work are being offered to the general public at no
charge under subsection 6d.

A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.
A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.
"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further
restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission
to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free
patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.
Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.
Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

1.30 awaitility 4.0.1
1.30.1 Available under license:

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

/*
 * Copyright 2019 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/Durations.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/LambdaErrorMessageGenerator.java

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

/*
 * Copyright 2014 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/LambdaErrorMessageGenerator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ConditionEvaluationHandler.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/AssertionCondition.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 374
Open Source Used In IOx-FogDirector-AC18 1.17.0

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

/*
 * Copyright 2008 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):

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

/*
 * Copyright (C) 2007 Google Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/spi/Timeout.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/EvaluationCleanup.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ExecutorLifecycle.java
No license file was found, but licenses were detected in source scan.
* Copyright 2009 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/reflect/FieldNameAndTypeMatcherStrategy.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/reflect/AssignableToFieldTypeMatcherStrategy.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/core/PredicateExceptionIgnorer.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/pollinterval/IterativePollInterval.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/core/DeadlockException.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/core/EvaluatedCondition.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
jar/org/awaitility/core/ExceptionIgnorer.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ConditionEvaluator.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ClassPathResolver.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ConditionTimeoutException.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/core/ThrowingRunnable.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/pollinterval/FibonacciPollInterval.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/pollinterval/HamcrestExceptionIgnorer.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/pollinterval/PollInterval.java
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/pollinterval/FixedPollInterval.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-jar/org/awaitility/Awaitility.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2010 PowerMock original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/core/CheckedExceptionRethrower.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041685_1614924779.91/0/awaitility-4-0-1-sources-1-
  jar/org/awaitility/core/ConditionEvaluationResult.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
1.31 libhcryptoeheimdal 7.5.0+dfsg-1

1.31.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Upstream-Name: heimdal

Upstream-Contact: heimdal@h5l.org

Source: https://github.com/heimdal/heimdal

Files-Excluded: doc/doxyout

lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *

Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California

License: BSD-3-clause

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in /usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#.
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
#.
# 1. The authors are not responsible for the consequences of use of this
#    software, no matter how awful, even if they arise from flaws in it.
#.
# 2. The origin of this software must not be misrepresented, either by
#    explicit claim or by omission. Since few users ever read sources,
#    credits must appear in the documentation.
#.
# 3. Altered versions must be plainly marked as such, and must not be
#    misrepresented as being the original software. Since few users
#    ever read sources, credits must appear in the documentation.
#.
# 4. This notice may not be removed or altered.
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

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

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

The parts of the libtelnet that handle Kerberos.

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may
require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

@end verbatim
@copynext

@heading The Regents of the University of California

The parts of the libroken, most of libtelnet, telnet, ftp, and popper.

@verbatim
Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
@heading TomsFastMath / LibTomMath

Tom’s fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim

@copynext

@heading Doug Rabson

GSS-API mechglue layer.

@verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@heading PADL Software Pty Ltd
@verbatim
Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PADL Software nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

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

@heading Marko Kreen

Fortuna in libhcrypto
Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@verbatim

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim

Copyright (c) 2006,2007
NTT (Nippon Telegraph and Telephone Corporation) . All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer as
   the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

@verbatim

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

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libroken

@verbatim
Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto
AES in libhcrypto

@verbatim

rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

This code is hereby placed in the public domain.

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

@end verbatim

@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim

Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@copynext

@heading Richard Outerbridge

DES core in libhcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

@end verbatim

@copynext

@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

* Neither the name of the Novell nor the names of its contributors may be used
to endorse or promote products derived from this software without specific
prior written permission.

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

@end verbatim

@end copyrightend

1.32 docker-java 3.2.5
1.32.1 Available under license :
   Apache-2.0

1.33 libaio 0.3.110 5ubuntu0.1
1.33.1 Available under license :
   Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
   Upstream-Name: libaio
   Upstream-Contact: Jeff Moyer <jmoyer@redhat.com>
   Source: https://pagure.io/libaio.git

   Files: *
   Copyright:
   Copyright 2002-2003 Benjamin LaHaise <bcrl@redhat.com>
   Copyright 2003-2004 Jeff Moyer <jmoyer@redhat.com>
   License: LGPL-2.1+
   This library is free software; you can redistribute it and/or
   modify it under the terms of the GNU Lesser General Public
   License as published by the Free Software Foundation; either
   version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library. If not, see <https://www.gnu.org/licenses/>.

Comment:
On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

^L

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility
is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work
      based on the Library, uncombined with any other library
      facilities. This must be distributed under the terms of the
      Sections above,

   b) Give prominent notice with the combined library of the fact
      that part of it is a work based on the Library, and explaining
      where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
1.34 libseccomp 2.5.1-1ubuntu1~18.04.2

1.34.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensor is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:
a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.
7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is
Copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. Because the library is licensed free of charge, there is no warranty for the library, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holders and/or other parties provide the library “as is” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the library is with you. Should the library prove defective, you assume the cost of all necessary servicing, repair or correction.

16. In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may modify and/or redistribute the library as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the library (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the library to operate with any other software), even if such holder or other party has been advised of the possibility of such damages.

1.35 berkeley-db 5.3.28-13.1ubuntu1.1

1.35.1 Available under license:

This package was debianized by Clint Adams <schizo@debian.org> on Wed, 25 Jul 2007 20:37:51 +0700

It was downloaded from http://www.oracle.com/technology/software/products/berkeley-db/db/index.html

Copyright and license:

The following is the license that applies to this copy of the Berkeley DB software. For a license to use the Berkeley DB software under conditions other than those described here, or to purchase support for this software, please contact Oracle at berkeleydb-info_us@oracle.com.

/*

*/
Copyright (c) 1990, 2010 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Redistributions in any form must be accompanied by information on how to obtain complete source code for the DB software and any accompanying software that uses the DB software. The source code must either be included in the distribution or be available for no more than the cost of distribution plus a nominal fee, and must be freely redistributable under reasonable conditions. For an executable file, complete source code means the source code for all modules it contains. It does not include source code for modules or files that typically accompany the major components of the operating system on which the executable file runs.

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

Copyright (c) 1990, 1993, 1994, 1995

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS `AS IS` AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*/
/*
* Copyright (c) 1995, 1996
* The President and Fellows of Harvard University. All rights reserved.
* *
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the University nor the names of its contributors
* may be used to endorse or promote products derived from this software
* without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY HARVARD AND ITS CONTRIBUTORS `AS IS` AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL HARVARD OR ITS CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*/
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

1.36 dnsjava 2.1.7

1.36.1 Available under license:

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.37 groovy 2.1.5

1.37.1 Available under license:

Found license 'Common Public License 1.0 (CPLv1.0)' in "* All rights reserved. The initial API is made available under the terms of * the Common Public License v1.0 which is available at:"

1.38 sigar 1.6.5.132-5

1.38.1 Available under license:

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

/*
* Copyright (c) 2006-2009 Hyperic, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

*/opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/SigarProxyCache.java
*/opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Version.java

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

/*
* Copyright (c) 2006-2009 Hyperic, Inc.
* Copyright (c) 2009 SpringSource, Inc.
* Copyright (c) 2010 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestPTQL.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006-2007 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/ptql/QueryLoadException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/ptql/StringPattern.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/vmware/VMwareServer.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/NetInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/win32/EventLog.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006-2007 Hyperic, Inc.
 * Copyright (c) 2010 VMware, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
http://www.apache.org/licenses/LICENSE-2.0

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

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/Pdh.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestProcTime.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Runner.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright (c) 2008 Hyperic, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Ls.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Pidof.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/FileVersion.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright (c) 2009 Hyperic, Inc.
* Copyright (c) 2009 SpringSource, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/ArchName.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2009 SpringSource, Inc.
* Copyright (c) 2010 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/Humidor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2006-2008 Hyperic, Inc.
* Copyright (c) 2009 SpringSource, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/SigarTestRunner.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2008-2009 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/FileVersionInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/win32/test/TestFileVersion.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006-2007 Hyperic, Inc.
 * Copyright (c) 2009 VMware, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
*/
Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/EventLogRecord.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2007, 2009 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarSwap.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/EventLogTail.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarRegistry.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarCpuInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarMem.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/AbstractMBean.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarLoadAverage.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarCpu.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/jmx/SigarCpuPerc.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestSignal.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2007-2008 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/test/TestNetStatPort.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/test/TestTcpStat.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2006 Hyperic, Inc.
* Copyright (c) 2010 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/cmd/Df.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2007 Hyperic, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestNfsClientV2.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestNfsServerV3.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/Nfsstat.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/ProcessQueryCompleter.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/ptql/SigarProcessQuery.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestNfsClientV3.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestNfsServerV2.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/win32/test/TestLocaleInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/win32/ProcFileMirror.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/win32/LocaleInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/jmx/CpuTimerMBean.java

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

/*
 * Copyright (c) 2009 Hyperic, Inc.
 * Copyright (c) 2009 SpringSource, Inc.
 * Copyright (c) 2009-2010 VMware, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/ArchNameTask.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2009 Hyperic, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/ArchLoader.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/ArchNotSupportedException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/ArchLoaderException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/jni/CopyDependsTask.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2006-2008 Hyperic, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/SigarLoader.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/CpuPerc.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/CpuInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/test/TestEventLog.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Iostat.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/MultiProcMem.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestNetInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/ServiceConfig.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/NfsFileSystem.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestProcStat.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/ptql/ProcessQueryFactory.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestNetIf.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/Win32.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/vmware/VMControlLibrary.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/cmd/Top.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/win32/Service.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/Sigar.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/test/TestNetStat.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/org/hyperic/sigar/MultiProcCpu.java

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

/*
 * Copyright (c) 2009 SpringSource, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestHumidor.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (c) 2006-2007, 2009 Hyperic, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/FileTail.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright (c) 2006, 2008 Hyperic, Inc.
* Copyright (c) 2010 VMware, Inc.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
Open Source Used In IOx-FogDirector-AC18 1.17.0  431

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestMem.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestProcState.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/Du.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/pager/PageList.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestProcModules.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/FileCompleter.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/ShellCommand_alias.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/ShellCommandInitException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/MemWatch.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/PagerProcessorExt.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/ShellCommand_alias.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/shell/ShellCommandInitException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/cmd/Ifconfig.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/pager/StaticPageFetcher.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/util/WeakReferenceMap.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/cmd/Uptime.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/shell/ShellCommandHandler.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/test/TestVMware.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/cmd/Win32Service.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/util/IteratorIterator.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/shell/ShellCommand_source.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/win32/EventLogThread.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/shell/ShellCommandExecException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/test/TestThreads.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/jmx/SigarInvokerJMX.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/win32/EventLogNotification.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/NfsUnreachableException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/test/GetPass.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/pager/PageFetcher.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/SigarNotImplementedException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/ProcEnv.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/shell/NormalQuitCommandException.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
org/hyperic/sigar/pager/DefaultPagerProcessor.java

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

*/
* Copyright (c) 2006-2007 Hyperic, Inc.
* Copyright (c) 2009 SpringSource, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/SigarProxy.java
No license file was found, but licenses were detected in source scan.

// ALL RIGHTS RESERVED
// Permission to use, copy, modify, and distribute this Software and its

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/util/PrintfFormat.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006-2007 Hyperic, Inc.
 * Copyright (c) 2009-2010 VMware, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/win32/test/TestPdh.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2006, 2008 Hyperic, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/OperatingSystem.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/SigarInvoker.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestInvoker.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestFileSystem.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestCpuInfo.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestProcEnv.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestWho.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestProcFd.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/SigarTestPrinter.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestRegistryKey.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestUptime.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestProcArgs.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/TestCpu.java

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

*/
* Copyright (c) 2006-2008 Hyperic, Inc.
* Copyright (c) 2010 VMware, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/ org/hyperic/sigar/cmd/Shell.java
* /opt/cola/permits/1004622435_1611680615.45/0/sigar-1-6-5-132-5-sources-jar/
  org/hyperic/sigar/test/SigarTestCase.java

1.39 libheimbaseheimdal 7.5.0+dfsg-1
1.39.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h51.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.
Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in
/usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
# .
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
# .
# 1. The authors are not responsible for the consequences of use of this
#    software, no matter how awful, even if they arise from flaws in it.
# .
# 2. The origin of this software must not be misrepresented, either by
#    explicit claim or by omission. Since few users ever read sources,
#    credits must appear in the documentation.
# .
# 3. Altered versions must be plainly marked as such, and must not be
#    misrepresented as being the original software. Since few users
#    ever read sources, credits must appear in the documentation.
# .
# 4. This notice may not be removed or altered.

@macro copynext{}
@vskip 20pt plus 1fil
@end macro
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

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

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

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

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

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@copynext
@heading The Regents of the University of California.

libedit

@verbatim
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

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

LibTomMath is hereby released into the Public Domain.

LibTomMath is hereby released into the Public Domain.

GSS-API mechglue layer.

Copyright (c) 2005 Doug Rabson
All rights reserved.

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

@end verbatim
@copynext

@heading PADL Software Pty Ltd

@verbatim
Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

@end verbatim
Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2006, 2007
NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer as
   the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libbroken

@verbatim

Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim
rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

This code is hereby placed in the public domain.

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

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c
Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Richard Outerbridge

DES core in libhcrypto

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec’s THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc.
All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

@end verbatim

@end verbatim

@verbatim

@end verbatim

@end verbatim

@end verbatim

@end verbatim

@end verbatim

@end verbatim
Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@end copyrightend

1.40 netty-codec-socks 4.1.49.Final

1.40.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 */
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksInitResponse.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socksx/v4/Socks4CommandType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/Socks5InitialRequest.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/SocksRequest.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/SocksAuthRequest.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/SocksAuthRequestDecoder.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/SocksCmdRequest.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/SocksCmdRequestDecoder.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1.jar/io/netty/handler/codec/socks/UnknownSocksRequest.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 453
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/META-INF/maven/io.netty/netty-codec-socks/pom.xml
No license file was found, but licenses were detected in source scan.

/
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socksx/v5/Socks5PasswordAuthRequestDecoder.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socksx/package-info.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socksx/v5/Socks5CommandResponseDecoder.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socksx/AbstractSocksMessage.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socks/SocksMessageType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/socks/SocksMessage.java
/* Copyright 2015 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

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

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

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

*/
*/
/* Copyright 2015 The Netty Project
*/
/* The Netty Project licenses this file to you under the Apache License,
*/
/* version 2.0 (the "License"); you may not use this file except in compliance
*/
/* with the License. You may obtain a copy of the License at:
*/
/*
*/
/* http://www.apache.org/licenses/LICENSE-2.0
*/
/*
*/
/* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.
*/
/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksCmdType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksAuthScheme.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksAddressType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/Socks5PasswordAuthStatus.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksResponseType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksAddressType.java
* /opt/ws_local/PERMITS_SQL/1068497245_1594417766.44/0/netty-codec-socks-4-1-49-final-sources-1-jar/io/netty/handler/codec/socks/SocksRequestType.java
*/

Open Source Used In IOx-FogDirector-AC18 1.17.0 456
1.41 javax-el 3.0.0

1.41.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation,
method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and
(b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the
terms of, this License. For legal entities,.You. includes any entity which controls, is controlled by, or is under
common control with You. For purposes of this definition,.control. means (a) the power, direct or indirect, to cause
the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty
percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use,
reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or
without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made,
use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or
otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the
Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the
combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,
each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use,
reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or
portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a
Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor
either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell,
offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions
thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or
portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients, rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.
You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION.

THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as
Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or
regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all
the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the
Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.
As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.42 junixsocket-native-common 2.3.2
1.42.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * junixsocket
 *
 * Copyright 2009-2020 Christian Kohlschütter
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):

*/opt/cola/permits/1258508010_1643078805.99/0/junixsocket-native-common-2-3-2-sources-jar/org/newsclub/lib/junixsocket/common/NarMetadata.java

1.43 javax-interceptor-api 1.2
1.43.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

   2.1. The Initial Developer Grant.

   Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.
3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the
terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara.
The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited
to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the
conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as
to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a
consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-
free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only
way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the
section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest
validity of any such claims; this section has the sole purpose of protecting the integrity of the free software
distribution system, which is implemented by public license practices. Many people have made generous
contributions to the wide range of software distributed through that system in reliance on consistent application of
that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other
system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted
interfaces, the original copyright holder who places the Program under this License may add an explicit
geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this
License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time
to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new
problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License
which applies to it and "any later version", you have the option of following the terms and conditions either of that
version or of any later version published by the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are
different, write to the author to ask for permission. For software which is copyrighted by the Free Software
Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be
guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE
PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM
"AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.44 hibernate-validator 6.1.6.Final
1.44.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2009 IIZUKA Software Technologies Ltd
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/TypeHelper.java
No license file was found, but licenses were detected in source scan.

/*
 * Hibernate Validator, declare and validate application constraints
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 */

Found in path(s):
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/ConcurrentReferenceHashMap.java
No license file was found, but licenses were detected in source scan.

~ Hibernate Validator, declare and validate application constraints
~
~ License: Apache License, Version 2.0
~ See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE

Found in path(s):
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/META-INF/validation-mapping-2.0.xsd
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/META-INF/validation-configuration-2.0.xsd
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/overview.html
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/META-INF/validation-configuration-1.0.xsd
/*
 * Hibernate Validator, declare and validate application constraints
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */
/**
 * Retrieves constraint related meta data for the parameters of the given
 * executable.
 *
 * @param javaBeanExecutable The executable of interest.
 *
 * @return A list with parameter meta data for the given executable.
 */

Found in path(s):
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
*/
expression, use its name as obtained by the active
{@link javax.validation.ParameterNameProvider}. The default provider will
return the actual parameter names, if the -parameters compiler option
has been enabled, and {@code arg0}, {@code arg1} etc. otherwise.

The following listing shows an example using the JavaScript engine which
comes with the JDK:

```java
@ParameterScriptAssert(script = "start.before(end)", lang = "javascript")
public void createEvent(Date start, Date end) { ... }
```

Can be specified on any method or constructor.

@since

@since

@author Gunnar Morling
*/

Found in path(s):
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/constraints/ParameterScriptAssert.java
No license file was found, but licenses were detected in source scan.

/*
 * Hibernate Validator, declare and validate application constraints
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */

Found in path(s):
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/number/sign/PositiveValidatorForNumber.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/HibernateValidatorPermission.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/Mod10CheckValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/valueextraction/ValueExtractorResolver.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/constraintvalidation/ComposingConstraintTree.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/time/past/PastValidatorForDate.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/SizeDef.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/ISBNValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/money/MinValidatorForMonetaryAmount.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/money/NegativeValidatorForMonetaryAmount.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/EANValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/br/CPFValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/hv/br/CPFValidator.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 483
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/valueextraction/ShortArrayValueExtractor.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/number/bound/decimal/DecimalMinValidatorForBigInteger.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/time/past/PastValidatorForMonthDay.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/MetaDataBuilder.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/rule/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/CollectionHelper.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/aggregated/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/facets/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metamodel/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/annotation/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/service/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/properties/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/valueextraction/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/messageinterpolation/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/valueextraction/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/propertyeditor/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/propertyeditor/PackageInfo.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/propertyeditor/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/context/CrossParameterConstraintMappingContext.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/notempty/NotEmptyValidatorForMap.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/context/PropertyConstraintMappingContext.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/validationcontext/ValidationContextBuilder.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/path/PropertyNode.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/constraints/time/DurationMax.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/util/InterpolationHelper.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/number/sign/PositiveValidatorForCharSequence.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/time/pastorpresent/AbstractPastOrPresentEpochBasedValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/number/bound/decimal/DecimalMinValidatorForBigDecimal.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/constraintvalidators/bv/number/sign/PositiveValidatorForByte.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/el/RootResolver.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/PatternDef.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/config/context/CrossParameterConstraintMappingContextImpl.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/privilegedactions/GetDeclaredConstructor.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/facets/Cascadable.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/pl/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/constraintvalidation/HibernateConstraintValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/pl/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/constraintvalidation/HibernateConstraintValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/pl/PESELDef.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/constraints/pl/NIP.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/messageinterpolation/DefaultLocaleResolver.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/engine/valueextraction/MapPropertyKeyExtractor.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/location/FieldConstraintLocation.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/descriptor/BeanDescriptorImpl.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/cfg/defs/CurrencyDef.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/util/TypeVariableBindings.java
* /opt/cola/permits/109933531_1602485629.53/0/hibernate-validator-6-1-6-final-sources-jar/org/hibernate/validator/internal/metadata/raw/ConstrainedType.java
jar/org/hibernate/validator/internal/constraintvalidators/bv/time/past/PastValidatorForYearMonth.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/number/bound/MinValidatorForBigInteger.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/spi/messageinterpolation/LocaleResolver.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/util/classhierarchy/package-info.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/cfg/defs/RangeDef.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/size/SizeValidatorForCharSequence.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/cfg/context/Cascadable.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/number/sign/PositiveValidatorForLong.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/constraints/Range.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/metadata/location/ConstraintLocation.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/constraintvalidation/HibernateConstraintValidatorContext.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/metadata/raw/AbstractConstrainedElement.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/EscapedState.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/time/AbstractFutureJavaTimeValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/number/bound/DecimalMaxValidatorForNumber.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/money/DigitsValidatorForMonetaryAmount.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/metadata/aggregated/ClassMetaData.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/engine/messageinterpolation/parser/EscapedState.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/constraintvalidation/HibernateConstraintValidatorContext.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/time/AbstractFutureJavaTimeValidator.java
* /opt/cola/permits/1099333531_1602486269.53/0/hibernate-validator-6-1-6-final-sources-
jar/org/hibernate/validator/internal/constraintvalidators/bv/time/PastOrPresentValidatorForYear.java
1.45 libgssapiheimdal 7.5.0+dfsg-1

1.45.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Please see info documentation for the complete list of licenses.

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the
pseudocode and C code), the author makes no guarantees and is not
responsible for any damage resulting from its use. The author grants
irrevocable permission to anyone to use, modify, and distribute it in
any way that does not diminish the rights of anyone else to use,
modify, and distribute it, provided that redistributed derivative
works do not contain misleading author or version information.
Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
License: GPL-2+

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in
/usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
#
# 1. The authors are not responsible for the consequences of use of this
   software, no matter how awful, even if they arise from flaws in it.
#
# 2. The origin of this software must not be misrepresented, either by
   explicit claim or by omission. Since few users ever read sources,
   credits must appear in the documentation.
#
# 3. Altered versions must be plainly marked as such, and must not be
   misrepresented as being the original software. Since few users
   ever read sources, credits must appear in the documentation.
#
# 4. This notice may not be removed or altered.

@macro copynext{ }
@vskip 20pt plus 1fil
@end macro

@macro copyrightstart{ }
@end macro
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

The parts of the libbroken, most of libtelnet, telnet, ftp, and popper.

Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@copynext

@heading The Regents of the University of California.

libedit

@verbatim

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

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

END VERBATIM
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@end copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim
@end copynext

@heading Doug Rabson

GSS-API mechglue layer.

@verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading PADL Software Pty Ltd

@table @asis
@item GSS-API CFX, SPNEGO, naming extensions, API extensions.
@item KCM credential cache.
@item HDB LDAP backend.
@end table

@verbatim

Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PADL Software nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim

Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim

Copyright (c) 2006,2007
NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libroken

@verbatim

Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

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

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

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

AES in libhcrypto

rijndael-alg-fst.c

version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

This code is hereby placed in the public domain.

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

Apple, Inc

kdc/announce.c

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@copynext

@heading Richard Outerbridge

DES core in libhcrypto

@verbatim

D3DES (V5.09)

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim
Windows support

Copyright (c) 2009-2015, Secure Endpoints Inc.
All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

Lib/hcrypto/test_dh.c

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

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

* Neither the name of the Novell nor the names of its contributors may be used
to endorse or promote products derived from this software without specific
prior written permission.

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

@end verbatim
@end copyrightend

1.46 libhxheimdal 7.5.0+dfsg-1

1.46.1 Available under license :

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Christos Zoulas of Cornell University.

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

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

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the
pseudocode and C code), the author makes no guarantees and is not
responsible for any damage resulting from its use. The author grants
irrevocable permission to anyone to use, modify, and distribute it in
any way that does not diminish the rights of anyone else to use,
modify, and distribute it, provided that redistributed derivative
works do not contain misleading author or version information.
Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
.
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
.
You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
.
On Debian systems, the full text can be found in
/usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
# .
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
# 1. The authors are not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.

# 2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources, credits must appear in the documentation.

# 3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.

# 4. This notice may not be removed or altered.

---

Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the Institute nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

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

@end verbatim
@copynext

@heading Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

@verbatim

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may
require a specific license from the United States Government.
It is the responsibility of any person or organization contemplating
export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and
distribute this software and its documentation for any purpose and
without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that copyright notice and
this permission notice appear in supporting documentation, and that
the name of M.I.T. not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior
permission. M.I.T. makes no representations about the suitability of
this software for any purpose. It is provided "as is" without express
or implied warranty.

@end verbatim
@copynext
The parts of the libroken, most of libtelnet, telnet, ftp, and popper.

Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Christos Zoulas of Cornell University.

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

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

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim

LibTomMath is hereby released into the Public Domain.

@end verbatim
@copynext

@heading Doug Rabson

GSS-API mechglue layer.
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim

Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim
Copyright (c) 2006, 2007
NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer as
the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libbroken
Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

@end verbatim
@copynext

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim
rijndael-alg-fst.c
@end verbatim

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

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

@verbatim

Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@verbatim

@end verbatim

@verbatim

@end verbatim

@end verbatim

@end verbatim
DES core in libhcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: David Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc. All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

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

@end verbatim

@copynext

@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

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

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

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

POSSIBILITY OF SUCH DAMAGE.

@end verbatim

@copyrightend

1.47 dpkg 1.19.0.5ubuntu2.3

1.47.1 Available under license:

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: dpkg

Files: *

Copyright:
Copyright 1994 Ian Murdock <imurdock@debian.org>
Copyright 1994 Matt Welsh <mdw@sunsite.unc.edu>
Copyright 1994 Carl Streeter <streeter@cae.wisc.edu>
Copyright 1994-1999, 2008 Ian Jackson <ijackson@chiark.greenend.org.uk>
Copyright 1995 Bruce Perens <bruce@pixar.com>
Copyright 1995-1996 Erick Branderhorst <branderhorst@heel.fgg.eur.nl>
Copyright 1996 Michael Shields <shields@crosslink.net>
Copyright 1996 Klee Dienes <klee@debian.org>
Copyright 1996 Kim-Minh Kaplan <kkkaplan@cdfhp3.in2p3.fr>
Copyright 1996-1998 Miquel van Smoorensburg <miquels@cistron.nl>
Copyright 1997-1998 Charles Briscoe-Smith <cpbs@debian.org>
Copyright 1997-1998 Juho Vuori <javuori@cc.helsinki.fi>
Copyright 1998 Koichi Sekido <sekids@mbox.kyoto-inet.or.jp>
Copyright 1998 Jim Van Zandt <jrv@vanzandt.mv.com>
Copyright 1998 Juan Cespedes <cespedes@debian.org>
Copyright 1998 Nilz Rennebarth <nils@debian.org>
Copyright 1998 Heiko Schlittermann <hs@sclittermann.de>
Copyright 1999 Roderick Shertler <roderick@argon.org>
Copyright 1999 Ben Collins <bcollins@debian.org>
Copyright 1999 Richard Kettlewell <rjk@sphere.greenend.org.uk>
Copyright 1999-2001 Marcus Brinkmann <brinkmd@debian.org>
Copyright 1999-2002 Wichert Akkerman <wakkerma@debian.org>
Copyright 2001, 2007, 2010 Joey Hess <joeyh@debian.org>
Copyright 2004-2005 Scott James Remnant <scott@netsplit.com>
Copyright 2006-2008 Frank Lichtenheld <djpig@debian.org>
Copyright 2006-2015 Guillem Jover <guillem@debian.org>
Copyright 2007-2012 Raphael Hertzog <hertzog@debian.org>
Copyright 2007 Nicolas Francois <nicolas.francois@centraliens.net>
Copyright 2007 Don Armstrong <don@donarmstrong.com>
Copyright 2007 Colin Watson <cjwatsen@debian.org>
Files:
lib/compat/getopt*
lib/compat/gettext.h
lib/compat/obstack.*
lib/compat/strnlen.c
Copyright:
Copyright 1987-2006 Free Software Foundation, Inc.
License: GPL-2+

Files:
dselect/methods/Dselect/Ftp.pm
dselect/methods/ftp/*
Copyright:
Copyright 1996 Andy Guy <awpguy@acs.ucalgary.ca>
Copyright 1998 Martin Schulze <joey@infodrom.org>
Copyright 1999-2001, 2005-2006, 2009 Raphal Hertzog <hertzog@debian.org>
License: GPL-2

Files:
scripts/Dpkg/Gettext.pm
Copyright:
Copyright 2000 Joey Hess <joeyh@debian.org>
Copyright 2007, 2009-2010, 2012-2015 Guillem Jover <guillem@debian.org>
License: BSD-2-clause

Files:
utils/start-stop-daemon.c
Copyright:
Copyright 1999 Marek Michalkiewicz <marekm@i17linuxb.ists.pwr.wroc.pl>
Copyright 1999 Christian Schwarz <schwarz@monet.m.isar.de>
Copyright 1999 Klee Dienes <klee@debian.org>
Copyright 1999 Ben Collins <bcollins@debian.org>
Copyright 1999-2002 Wichert Akkerman <wakkerma@debian.org>
Copyright 2000-2003 Adam Heath <doogie@debian.org>
Copyright 2001 Sontri Tomo Huynh <huynh.29@osu.edu>
Copyright 2001 Andreas Schuldei <andreas@schuldei.org>
Copyright 2001 Ian Jackson <ijackson@chiark.greenend.org.uk>
Copyright 2004-2005 Scott James Remnant <keybuk@debian.org>
Copyright 2006-2014 Guillem Jover <guillem@debian.org>
Copyright 2008 Samuel Thibault <samuel.thibault@ens-lyon.org>
Copyright 2008 Andreas Phlsson <andreas.pahlsson@xcerion.com>
Copyright 2009 Chris Coulson <chriscoulson@googlemail.com>
Copyright 2012 Carsten Hey <carsten@debian.org>
Copyright 2014 Nir Soffer <nirs@hyperms.com>
License: public-domain-s-s-d
Written by Marek Michalkiewicz <marekm@i17linuxb.ists.pwr.wroc.pl>,
public domain. Based conceptually on start-stop-daemon.pl, by Ian
Jackson <ijackson@gnu.ai.mit.edu>. May be used and distributed
freely for any purpose. Changes by Christian Schwarz
<schwarz@monet.m.isar.de>, to make output conform to the Debian
Console Message Standard, also placed in public domain. Minor
changes by Klee Dienes <klee@debian.org>, also placed in the Public
Domain.

Changes by Ben Collins <bcollins@debian.org>, added --chuid, --background
and --make-pidfile options, placed in public domain as well.

Files: lib/compat/md5.*
Copyright:
Copyright 1993 Colin Plumb
License: public-domain-md5
This code implements the MD5 message-digest algorithm.
The algorithm is due to Ron Rivest. This code was
written by Colin Plumb in 1993, no copyright is claimed.
This code is in the public domain; do with it what you wish.

Equivalent code is available from RSA Data Security, Inc.
This code has been tested against that, and is equivalent,
except that you don't need to include two pages of legalese
with every copy.

License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.
Comment:
On Debian systems, the complete text of the GNU General Public License
can be found in /usr/share/common-licenses/GPL-2 or in the dpkg source
as the file COPYING.
License: GPL-2
This is free software; you can redistribute it and/or modify
it under the terms of version 2 of the GNU General Public
License version 2 as published by the Free Software Foundation.

This is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.

License: BSD-2-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.48 qemu-block-extra 2.11+dfsg-1ubuntu7.39
1.48.1 Available under license :
/
* FreeBSD strace list
* *
* This program is free software; you can redistribute it and/or modify

Open Source Used In IOx-FogDirector-AC18 1.17.0 555
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program; if not, see <http://www.gnu.org/licenses/>.
*/

{ TARGET_FREEBSD_NR___acl_aclcheck_fd, "__acl_aclcheck_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_aclcheck_file, "__acl_aclcheck_file", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_aclcheck_link, "__acl_aclcheck_link", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_fd, "__acl_delete_fd", "%s(%d, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_file, "__acl_delete_file", "%s("%s"", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_link, "__acl_delete_link", "%s("%s"", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_fd, "__acl_get_fd", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_file, "__acl_get_file", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_link, "__acl_get_link", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_fd, "__acl_set_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_file, "__acl_set_file", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_link, "__acl_set_link", "%s("%s"", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___semctl, "__semctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___syscall, "__syscall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___syscti, "__syscti", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___umtx_op, "__umtx_op", "%s(%#x, %d, %d, %#x, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___accept, "accept", "%s(%d,%#x,%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___access, "access", "%s("%s",%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR___acct, "acct", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___adjtime, "adjtime", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___bind, "bind", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___break, "break", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chdir, "chdir", "%s("%s")", NULL, NULL },
{ TARGET_FREEBSD_NR___chflags, "chflags", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chmod, "chmod", "%s("%s",%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR___chown, "chown", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chroot, "chroot", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_getres, "clock_getres", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_gettime, "clock_gettime", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_settime, "clock_settime", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___close, "close", "%s(%d)", NULL, NULL },
{ TARGET_FREEBSD_NR___connect, "connect", "%s(%d,%#x,%d)", NULL, NULL },
{ TARGET_FREEBSD_NR___dup, "dup", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___dup2, "dup2", NULL, NULL, NULL },

Open Source Used In IOx-FogDirector-AC18 1.17.0 556
{ TARGET_FREEBSD_NR_getlogin, "getlogin", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getpeername, "getpeername", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getpgid, "getpgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getpgrp, "getpgrp", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_getpid, "getpid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_getppid, "getppid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_getpriority, "getpriority", "%s(%#x,%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_getresgid, "getresgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getresuid, "getresuid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_getrlimit, "getrlimit", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getrusage, "getrusage", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getsid, "getsid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getsockname, "getsockname", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getsockopt, "getsockopt", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_gettimeofday, "gettimeofday", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getuid, "getuid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_ioctl, "ioctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_issetugid, "issetugid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_kevent, "kevent", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_kill, "kill", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_kqueue, "kqueue", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ktrace, "ktrace", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_lchown, "lchown", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_link, "link", "%s("%s","%s")", NULL, NULL },
{ TARGET_FREEBSD_NR_lseek, "lseek", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_lstat, "lstat", "%s("%s",%p)", NULL, NULL },
{ TARGET_FREEBSD_NR_lpathconf, "lpathconf", "%s("%s", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR_lseek, "lseek", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_madvise, "madvise", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mincore, "mincore", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mkdir, "mkdir", "%s("%s",%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR_mkfifo, "mkfifo", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mknod, "mknod", "%s("%s",%#o,%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_mlock, "mlock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mlockall, "mlockall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mmap, "mmap", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msgctl, "msgctl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msgget, "msgget", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msgrcv, "msgrcv", "msgretriever", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_msync, "msync", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munlock, "munlock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munlockall, "munlockall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_munmap, "munmap", "%s(%p,%d)", NULL, NULL },
{ TARGET_FREEBSD_NR_nanosleep, "nanosleep", NULL, NULL, NULL },

Open Source Used In IOx-FogDirector-AC18 1.17.0 558
{ TARGET_FREEBSD_NR_nfssvc, "nfssvc", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_open, "open", "%s\n%\s\n%\#x, %\#o\n", NULL, NULL },
{ TARGET_FREEBSD_NR_openat, "openat", "%s\n%\d, %\s\n%\#x, %\#o\n", NULL, NULL },
{ TARGET_FREEBSD_NR_pathconf, "pathconf", "%s\n%\s\n%\d\n", NULL, NULL },
{ TARGET_FREEBSD_NR_pipe, "pipe", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_poll, "poll", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pread, "pread", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_preadv, "preadv", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_profif, "profif", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ptrace, "ptrace", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwrite, "pwrite", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwritev, "pwritev", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_quotactl, "quotactl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_read, "read", "%s\n%\d, %\#x, %\d\n", NULL, NULL },
{ TARGET_FREEBSD_NR_readlink, "readlink", "%s\n%\s\n%\#p, %\#d\n", NULL, NULL },
{ TARGET_FREEBSD_NR_reboot, "reboot", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_recvfrom, "recvfrom", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_recvmsg, "recvmsg", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rename, "rename", "%s\n%\s\n%\s\n%\s\n", NULL, NULL },
{ TARGET_FREEBSD_NR_revoke, "revoke", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rfork, "rfork", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rmdir, "rmdir", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_rtprio_thread, "rtprio_thread", "%s\n%\d, %\d, %\p\n", NULL, NULL },
{ TARGET_FREEBSD_NR_sbrk, "sbrk", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sched_yield, "sched_yield", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_select, "select", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_semget, "semget", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_semop, "semop", "%s\n%\d, %\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_sendmsg, "sendmsg", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_sendto, "sendto", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setcontext, "setcontext", "%s\n%\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_setegid, "setegid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_seteuid, "seteuid", "%s\n%\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_setgid, "setgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setgroups, "setgroups", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setitimer, "setitimer", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setlogin, "setlogin", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setpgid, "setpgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setpriority, "setpriority", "%s\n%\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_setreuid, "setreuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_setsid, "setsid", "%s\n%\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_setsockopt, "setsockopt", "%s\n%\d, %\#x\n", NULL, NULL },
{ TARGET_FREEBSD_NR_settimeofday, "settimeofday", NULL, NULL, NULL },
}
NOTE! This copyright does *not* cover user programs that use kernel services by normal system calls - this is merely considered normal use of the kernel, and does *not* fall under the heading of "derived work". Also note that the GPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the Linux kernel) is copyrighted by me and others who actually wrote it. Also note that the only valid version of the GPL as far as the kernel
is concerned is _this_ particular version of the license (ie v2, not
v2.2 or v3.x or whatever), unless explicitly otherwise stated.

Linus Torvalds

----------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
{Open Source Used In IOx-FogDirector-AC18 1.17.0 561}
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program ‘Gnomovision’ (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
Slirp was written by Danny Gasparovski.
Copyright (c), 1995,1996 All Rights Reserved.

Slirp is maintained by Kelly Price <tygris+slirp@erols.com>

Slirp is free software; "free" as in you don't have to pay for it, and you are free to do whatever you want with it. I do not accept any donations, monetary or otherwise, for Slirp. Instead, I would ask you to pass this potential donation to your favorite charity. In fact, I encourage *everyone* who finds Slirp useful to make a small donation to their favorite charity (for example, GreenPeace). This is not a requirement, but a suggestion from someone who highly values the service they provide.

The copyright terms and conditions:

---BEGIN---

Copyright (c) 1995,1996 Danny Gasparovski. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

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

---END---

This basically means you can do anything you want with the software, except
1) call it your own, and 2) claim warranty on it. There is no warranty for
this software. None. Nada. If you lose a million dollars while using
Slirp, that's your loss not mine. So, ***USE AT YOUR OWN RISK***.

If these conditions cannot be met due to legal restrictions (E.g. where it
is against the law to give out Software without warranty), you must cease
using the software and delete all copies you have.

Slirp uses code that is copyrighted by the following people/organizations:

Juha Pirkola.
Gregory M. Christy.
The Regents of the University of California.
Carnegie Mellon University.
The Australian National University.
RSA Data Security, Inc.

Please read the top of each source file for the details on the various
copyrights.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
      stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
      whole or in part contains or is derived from the Program or any
      part thereof, to be licensed as a whole at no charge to all third
      parties under the terms of this License.

   c) If the modified program normally reads commands interactively
      when run, you must cause it, when started running for such
      interactive use in the most ordinary way, to print or display an
      announcement including an appropriate copyright notice and a
      notice that there is no warranty (or else, saying that you provide
      a warranty) and that users may redistribute the program under
      these conditions, and telling the user how to view a copy of this
      License. (Exception: if the Program itself is interactive but
      does not normally print such an announcement, your work based on
      the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANDABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
Copyright (c) Individual contributors.
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

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

3. Neither the name of PyCA Cryptography nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

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

The following points clarify the QEMU license:

1) QEMU as a whole is released under the GNU General Public License, version 2.

2) Parts of QEMU have specific licenses which are compatible with the GNU General Public License, version 2. Hence each source file contains its own licensing information. Source files with no licensing information are released under the GNU General Public License, version 2 or (at your option) any later version.

As of July 2013, contributions under version 2 of the GNU General Public License (and no later version) are only accepted for the following files or directories: bsd-user/, linux-user/, hw/vfio/, hw/xen/xen_pt*.

3) The Tiny Code Generator (TCG) is released under the BSD license (see license headers in files).

4) QEMU is a trademark of Fabrice Bellard.

Fabrice Bellard and the QEMU team

A. HISTORY OF THE SOFTWARE

Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see http://www.cnri.reston.va.us) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations (now Zope Corporation, see http://www.zope.com). In 2001, the Python Software Foundation (PSF, see http://www.python.org/psf/) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation is a sponsoring member of the PSF.
All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td></td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
</tr>
<tr>
<td>1.3 thru 1.5.2 1.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2</td>
<td>2.1.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.1</td>
<td>2.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.2</td>
<td>2.2.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.3</td>
<td>2.2.2</td>
<td>2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3</td>
<td>2.2.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.1</td>
<td>2.3</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.2</td>
<td>2.3.1</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.3</td>
<td>2.3.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.4</td>
<td>2.3.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.5</td>
<td>2.3.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4</td>
<td>2.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.1</td>
<td>2.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.2</td>
<td>2.4.1</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.3</td>
<td>2.4.2</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5</td>
<td>2.4</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.7</td>
<td>2.6</td>
<td>2010</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.
Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
-------------------------------------------

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publically, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture
between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
---------------------------------------

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2
--------------------------------------------------

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior
permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.
For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has...
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

LICENCE
=======

The software in this repository is covered by the following licence.

// Copyright 2013, ARM Limited
// All rights reserved.
// Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
// * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
// * Neither the name of ARM Limited nor the names of its contributors may be...
// used to endorse or promote products derived from this software without
// specific prior written permission.

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

All the files in this directory and subdirectories are released under
a BSD like license (see header in each file). No other license is
accepted.
argparse is (c) 2006-2009 Steven J. Bethard <steven.bethard@gmail.com>.

The argparse module was contributed to Python as of Python 2.7 and thus
was licensed under the Python license. Same license applies to all files in
the argparse package project.

For details about the Python License, please see doc/Python-License.txt.

History
-------

Before (and including) argparse 1.1, the argparse package was licensed under
Apache License v2.0.

After argparse 1.1, all project files from the argparse project were deleted
due to license compatibility issues between Apache License 2.0 and GNU GPL v2.

The project repository then had a clean start with some files taken from
Python 2.7.1, so definitely all files are under Python License now.

1.49 commons-configuration 1.5

1.49.1 Available under license:

Apache Commons Configuration
Copyright 2001-2007 The Apache Software Foundation

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

Version 2.0, January 2004
http://www.apache.org/licenses/
 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.50 xerces-j 2.7.1
1.50.1 Available under license:

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.51 spring-tx 5.2.9.RELEASE
1.51.1 Available under license:

   Spring Framework 5.2.9.RELEASE
   Copyright (c) 2002-2020 Pivotal, Inc.

   This product is licensed to you under the Apache License, Version 2.0
   (the "License"). You may not use this product except in compliance with
   the License.

   This product may include a number of subcomponents with separate
   copyright notices and license terms. Your use of the source code for
   these subcomponents is subject to the terms and conditions of the
   subcomponent's license, as noted in the license.txt file.

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

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

=======================================================================

SPRING FRAMEWORK 5.2.9.RELEASE SUBCOMPONENTS:

Spring Framework 5.2.9.RELEASE includes a number of subcomponents
with separate copyright notices and license terms. The product that
includes this file does not necessarily use all the open source
subcomponents referred to below. Your use of the source
code for these subcomponents is subject to the terms and
conditions of the following licenses.


Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

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

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

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

3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
 LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.
>>> CGLIB 3.3 (cglib:cglib:3.3):

Per the LICENSE file in the CGLIB JAR distribution downloaded from https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar, CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which is included above.

>>> Objenesis 3.1 (org.objenesis:objenesis:3.1):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html, Objenesis 3.1 is licensed under the Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html and corresponding to section 4d of the Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

===============================================================================

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.
1.52 shiro-crypto-cipher 1.7.1

1.52.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

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

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

1.53 ubuntu-keyring 2018.09.18.1~18.04.2

1.53.1 Available under license:

This is Ubuntu GNU's GnuPG keyrings of archive keys.

This package was originally put together by Michael Vogt
<michael.vogt@canonical.com>

The keys in the keyrings don't fall under any copyright. Everything else in the package is covered by the GNU GPL.

Ubuntu support files Copyright (C) 2004 Michael Vogt <michael.vogt@canonical.com> based on the debian-keyring package maintained by James Troup

Ubuntu support files for ubuntu-keyring are free software; you can redistribute them and/or modify them under the terms of the GNU General Public License as published by the Free Software Foundation;
either version 2, or (at your option) any later version.

Ubuntu support files for ubuntu-keyring are distributed in the hope that they will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License with your Ubuntu system, in /usr/share/common-licenses/GPL, or with the Ubuntu GNU ubuntu-keyring source package as the file COPYING. If not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

1.54 jetty-plus 9.4.31.v20200723

1.54.1 Available under license:

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

Eclipse Public License - v 1.0

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

1. DEFINITIONS

"Contribution" means:

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

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

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

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

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

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

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

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

3. REQUIREMENTS

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

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

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

4. COMMERCIAL DISTRIBUTION

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

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

5. NO WARRANTY

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

6. DISCLAIMER OF LIABILITY

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

7. GENERAL

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

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

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

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

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

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

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

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

-----

Oracle

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

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

-----

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

* java.sun.security.ssl

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

-------

OW2

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

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

-------

Apache

The following artifacts are ASL2 licensed.

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

-------

MortBay

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

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

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

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

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

1.55 hdrhistogram 2.1.12
1.55.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 */

* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/WriterReaderPhaser.java
No license file was found, but licenses were detected in source scan.

/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 * @author Gil Tene
 */

* /opt/cola/permits/1257203437_1642787675.09/0(hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/DoublePercentileIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/ZigZagEncoding.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/AllValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/Base64Helper.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/IntCountsHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/DoubleLinearIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/PackedConcurrentDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/SynchronizedHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/packedarray/PackedArraySingleWriterRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/Histogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/LogarithmicIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/Recorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/ConcurrentDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/HistogramLogReader.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/HistogramLogProcessor.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/PackedHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/HistogramIterationValue.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/AtomicHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/SingleWriterDoubleRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/DoubleRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-
jar/org/HdrHistogram/DoubleHistogramIterationValue.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AbstractHistogramIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/SynchronizedDoubleHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/PercentileIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/PackedConcurrentHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleRecordedValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleAllValuesIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/SingleWriterRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleLogarithmicIterator.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ShortCountsHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/EncodableHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/HistogramLogScanner.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/packedarray/PackedArrayRecorder.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/ConcurrentHistogram.java

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

/*
* Written by Gil Tene of Azul Systems, and released to the public domain,
* as explained at http://creativecommons.org/publicdomain/zero/1.0/
* *
* @author Gil Tene
*/

Found in path(s):
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/AbstractHistogram.java
* /opt/cola/permits/1257203437_1642787675.09/0/hdrhistogram-2-1-12-sources-jar/org/HdrHistogram/DoubleHistogram.java

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

/*
* package-info.java
* Written by Gil Tene of Azul Systems, and released to the public domain,
* as explained at http://creativecommons.org/publicdomain/zero/1.0/
*/
No license file was found, but licenses were detected in source scan.

```java
/**
 * Written by Gil Tene of Azul Systems, and released to the public domain,
 * as explained at http://creativecommons.org/publicdomain/zero/1.0/
 *
 * @author Gil Tene
 */

package org.HdrHistogram;

final class Version {
    public static final String version="$VERSION$";
    public static final String build_time="$BUILD_TIME$";
}
```

1.56 libsemanage 2.7 2build2

1.56.1 Available under license:

```
GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
courage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating
system.

Although the Lesser General Public License is Less protective of the
users' freedom, it does ensure that the user of a program that is
linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, whereas the latter must
be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
“any later version”, you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
That's all there is to it!
This is the Debian package for libsemanage, and it is built from sources

libsemanage is Copyright 2004-2007 Tresys Technology, LLC
Copyright 2005 Red Hat, Inc.

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1

On Debian GNU/Linux systems, the complete text of the Lesser GNU General
Public License can be found in `/usr/share/common-licenses/LGPL'.

This package is maintained by Manoj Srivastava <srivasta@debian.org>.

The Debian specific changes are 2005-2009, Manoj Srivastava
<srivasta@debian.org>, and distributed under the terms of the GNU
General Public License, version 2.

On Debian GNU/Linux systems, the complete text of the GNU General
Public License can be found in `/usr/share/common-licenses/GPL'.

A copy of the GNU General Public License is also available at
<URL:http://www.gnu.org/copyleft/gpl.html>. You may also obtain
it by writing to the Free Software Foundation, Inc., 51 Franklin
St, Fifth Floor, Boston, MA 02110-1301, USA.

Manoj Srivastava <srivasta@debian.org>
arch-tag: d4250e44-a0e0-4ee0-adb9-2bd74f6e2eb7

1.57 jetty-io 9.4.31.v20200723
1.57.1 Available under license:

This program and the accompanying materials are made available under the
terms of the Eclipse Public License 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0, or the Apache Software License
Eclipse Public License - v 1.0

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

1. DEFINITIONS

"Contribution" means:

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

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

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

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

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

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

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

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

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

3. REQUIREMENTS

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

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

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

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

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

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

4. COMMERCIAL DISTRIBUTION

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

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

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

5. NO WARRANTY

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

6. DISCLAIMER OF LIABILITY

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

7. GENERAL

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

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

All Recipient’s rights under this Agreement shall terminate if it fails to
comply with any of the material terms or conditions of this Agreement and does
not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate,
Recipient agrees to cease use and distribution of the Program as soon as
reasonably practicable. However, Recipient's obligations under this Agreement
and any licenses granted by Recipient relating to the Program shall continue
and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be
modified in the following manner. The Agreement Steward reserves the right to
publish new versions (including revisions) of this Agreement from time to
time. No one other than the Agreement Steward has the right to modify this
Agreement. The Eclipse Foundation is the initial Agreement Steward. The
Eclipse Foundation may assign the responsibility to serve as the Agreement
Steward to a suitable separate entity. Each new version of the Agreement will
be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

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

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

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.
Eclipse

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

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

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

------

Oracle

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

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

------

Oracle OpenJDK

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

* java.sun.security.ssl

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

------

OW2

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

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

Apache

The following artifacts are ASL2 licensed.

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

-----

MortBay

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

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

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

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----

Assorted

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

1.58 httpcomponents-mime 4.5.13
1.58.1 Available under license:

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work.
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
   distribution, then any Derivative Works that You distribute must
   include a readable copy of the attribution notices contained
   within such NOTICE file, excluding those notices that do not
   pertain to any part of the Derivative Works, in at least one
   of the following places: within a NOTICE text file distributed
   as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

========================================================================
This project includes Public Suffix List copied from <https://publicsuffix.org/list/effective_tld_names.dat>
Mozilla Public License Version 2.0
==================================

1. Definitions
------------

1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described
   in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of
   version 1.1 or earlier of the License, but not also under the
terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation
Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for
the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,    *
*  repair, or correction. This disclaimer of warranty constitutes an    *
*  essential part of this License. No use of any Covered Software is    *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
* Under no circumstances and under no legal theory, whether tort *
* (including negligence), contract, or otherwise, shall any *
* Contributor, or anyone who distributes Covered Software as *
* permitted above, be liable to You for any direct, indirect, *
* special, incidental, or consequential damages of any character *
* including, without limitation, damages for lost profits, loss of *
* goodwill, work stoppage, computer failure or malfunction, or any *
* and all other commercial damages or losses, even if such party *
* shall have been informed of the possibility of such damages. This *
* limitation of liability shall not apply to liability for death or *
* personal injury resulting from such party's negligence to the *
* extent applicable law prohibits such limitation. Some *
* jurisdictions do not allow the exclusion or limitation of *
* incidental or consequential damages, so this exclusion and *
* limitation may not apply to You. *
*  
*   ******************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous
-----------------

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License
----------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.
10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
1.59.1 Available under license:

/*
 * Copyright (c) 1989 The Regents of the University of California.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *       This product includes software developed by the University of
 *       California, Berkeley and its contributors.
 * 4. Neither the name of the University nor the names of its contributors
 *    may be used to endorse or promote products derived from this software
 *    without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED.  IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 */

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later
version.

The complete text of the license is available in the
../Documentation/licenses/COPYING.LGPLv2.1 file.

WEV @@ WEV[B "1

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),

Open Source Used In IOx-FogDirector-AC18 1.17.0 667
conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This library is free software; you can redistribute it and/or modify it under the terms of the Modified BSD License. The complete text of the license is available in the../Documentation/licenses/COPYING.BSD-3 file.
1.60 jsr166e 1.0

1.60.1 Available under license:

Public Domain
http://creativecommons.org/publicdomain/zero/1.0/

1.61 commons-io 2.4

1.61.1 Available under license:

Apache Commons IO
Copyright 2002-2012 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.62 spring-data-jpa 2.1.5.RELEASE

1.62.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all...
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=======================================================================

To the extent any open source subcomponents are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by
downloading the Source Files from http://www.springsource.org/download, or by sending a request, with your name and address to: VMware, Inc., 3401 Hillview Avenue, Palo Alto, CA 94304, United States of America or email info@vmware.com. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. VMware shall mail a copy of the Source Files to you on a CD or equivalent physical medium. This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product.

Spring Data JPA 2.1.5
Copyright (c) [2011-2019] Pivotal Software, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE file.

1.63 netty-transport-native-unix-common
4.1.46.Final
1.63.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/IovArray.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/UnixChannelOption.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-
No license file was found, but licenses were detected in source scan.

~ Copyright 2016 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4.1.46-final-sources-jar/io/netty/channel/unix/UnixChannelUtil.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4.1.46-final-sources-jar/netty_unix_ini.h

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 */
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
* /

Found in path(s):
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_buffer.c
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/PreferredDirectByteBufAllocator.java
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/Buffer.java
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_buffer.h
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2015 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
* /

Found in path(s):
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_socket.c
  * /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/FileDescriptor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/DatagramSocketAddress.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/UnixChannel.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_socket.h
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_filedescriptor.c
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_filedescriptor.h
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_errors.h
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/DomainSocketReadMode.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/netty_unix_errors.c
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
* /opt/cola/permits/1283432436_1646722314.37/0/netty-transport-native-unix-common-4-1-46-final-sources-jar/io/netty/channel/unix/Errors.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 684
1.64 dash 0.5.8 2.10
1.64.1 Available under license:
MIT License

Copyright (c) 2017 Mike Wilcox

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.65 log4j-jcl 2.17.0
1.65.1 Available under license:

Apache Log4j Commons Logging Bridge
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.66 metrics-annotation 3.1.2
1.66.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2012 Ryan W Tenney (ryan@10e.us)
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432582_1646721953.19/0/metrics-annotation-3-1-2-sources-
  jar/com/codahale/metrics/annotation/Metric.java

1.67 lz4 0.0~r131-2ubuntu3.1
1.67.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Copyright (c) 2014, lpsantil
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

LZ4 Library
Copyright (c) 2011-2014, Yann Collet
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

### 1.68 shiro-config-ogdl 1.7.1

#### 1.68.1 Available under license:

**Apache Shiro**

Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based on initial ideas from Dr. Heinz Kabutz's publicly posted version available at http://www.javaspecialists.eu/archive/Issue015.html, with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

**Apache License**

Version 2.0, January 2004

http://www.apache.org/licenses/

**TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION**

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.69 libcap-ng 0.7.7 3.1

1.69.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations
below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user’s freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.
Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

^L

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

## 1.70 mockito 1.10.19

### 1.70.1 Available under license:

Mockito license - MIT.

Libraries used:

- Cglib - Apache License 2.0
- ASM - BSD license

Mockito all distribution:

- Objenesis - MIT license
- Hamcrest - BSD license

Copyright (c) 2000-2005 INRIA, France Telecom

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

Open Source Used In IOx-FogDirector-AC18 1.17.0 727
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.71 cglib 3.3.0
1.71.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.72 gson 2.8.0

1.72.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.73 udev 241-7~deb10u3

1.73.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

   Gnomovision version 69, Copyright (C) year name of author
   Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
   This is free software, and you are welcome to redistribute it
   under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

   Yoyodyne, Inc., hereby disclaims all copyright interest in the program
   `Gnomovision' (which makes passes at compilers) written by James Hacker.

   <signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.74 hazelcast-kubernetes 1.0.0

1.74.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2008-2017, Hazelcast, Inc. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432902_1646723438.99/0/hazelcast-kubernetes-1-0-0-sources-jar/com/hazelcast/kubernetes/ServiceEndpointResolver.java
* /opt/cola/permits/1283432902_1646723438.99/0/hazelcast-kubernetes-1-0-0-sources-jar/com/hazelcast/kubernetes/DnsEndpointResolver.java
* /opt/cola/permits/1283432902_1646723438.99/0/hazelcast-kubernetes-1-0-0-sources-jar/com/hazelcast/kubernetes/KubernetesProperties.java
* /opt/cola/permits/1283432902_1646723438.99/0/hazelcast-kubernetes-1-0-0-sources-jar/com/hazelcast/kubernetes/package-info.java

No license file was found, but licenses were detected in source scan.

- Licensed under the Apache License, Version 2.0 (the "License");
- you may not use this file except in compliance with the License.
- You may obtain a copy of the License at
Found in path(s):
No license file was found, but licenses were detected in source scan.

# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
com.hazelcast.kubernetes.HazelcastKubernetesDiscoveryStrategyFactory

Found in path(s):

1.75 activation-api 1.2.0

1.75.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than
Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.
2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or
otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.
4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding
declaratory judgment actions) against Initial Developer or a
Contributor (the Initial Developer or Contributor against whom You
assert such claim is referred to as "Participant") alleging that the
Participant Software (meaning the Contributor Version where the
Participant is a Contributor or the Original Software where the
Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or
indirectly to You by such Participant, the Initial Developer (if the
Initial Developer is not the Participant) and all Contributors under
Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice
from Participant terminate prospectively and automatically at the
expiration of such 60 day notice period, unless if within such 60
day period You withdraw Your claim with respect to the Participant
Software against such Participant either unilaterally or pursuant to
a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant
alleging that the Participant Software directly or indirectly
infringes any patent where such claim is resolved (such as by
license or settlement) prior to the initiation of patent
infringement litigation, then the reasonable value of the licenses
granted by such Participant under Sections 2.1 or 2.2 shall be taken
into account in determining the amount or value of any payment or
license.

6.4. In the event of termination under Sections 6.1 or 6.2 above,
all end user licenses that have been validly granted by You or any
distributor hereunder prior to termination (excluding licenses
granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT
(INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE
INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF
COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE
TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT
LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER
FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR
LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE
POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT
APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH
PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH
LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR
LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES. SO THIS EXCLUSION
AND LIMITATION MAY NOT APPLY TO YOU.
8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

------------------------------------------------------------------------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code.
And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new
versions of the General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingling code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a
lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.76 annotations 13.0

1.77 libkrbheimdal 7.5.0+dfsg-1
1.77.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software.
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Högskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in /usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
#
# 1. The authors are not responsible for the consequences of use of this
#    software, no matter how awful, even if they arise from flaws in it.
#
# 2. The origin of this software must not be misrepresented, either by
#    explicit claim or by omission. Since few users ever read sources,
#    credits must appear in the documentation.
#

Open Source Used In IOx-FogDirector-AC18 1.17.0 767
# 3. Altered versions must be plainly marked as such, and must not be
# misrepresented as being the original software. Since few users
# ever read sources, credits must appear in the documentation.
#
# 4. This notice may not be removed or altered.

@macro copynext{}
@end macro

@vskip 20pt plus 1fil
@end macro

@macro copyrightstart{}
@end macro

@macro copyrightend{}
@end macro

@node Copyrights and Licenses, , Acknowledgments, Top
@comment node-name, next, previous, up
@appendix Copyrights and Licenses

@heading Kungliga Tekniska Hgskolan

@copyrightstart
@verbatim
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE 
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE 
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE 
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL 
 DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS 
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) 
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT 
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY 
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF 
SUCH DAMAGE.

@end verbatim
@copynext
@heading Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

@end verbatim
@copynext
@heading The Regents of the University of California

The parts of the libroken, most of libtelnet, telnet, ftp, 
and popper.

@end verbatim
Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext

@heading The Regents of the University of California.

libedit

@verbatim
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS \`AS IS\' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim
@copynext

@heading Doug Rabson

GSS-API mechglue layer.

@verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@verbatim
@end verbatim
@copynext

@heading PADL Software Pty Ltd

@verbatim
Copyrigh (c) 2003-2011, PADL Software Pty Ltd.
Copyrig (c) 2004, Andrew Bartlett.
Copyrig (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyrig (c) 2015, Timothy Pearson.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
@end verbatim
3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim
Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

END VERBATIM
@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim

Copyright (c) 2006,2007
NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer as
the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libbroken

@verbatim

Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim
rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
Open Source Used In IOx-FogDirector-AC18 1.17.0 776

WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim
Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim
@copynext

@heading Richard Outerbridge

DES core in libhcrypto

@verbatim
D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.
Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
1.78 stax-ex 1.8

1.78.1 Available under license:

Found license 'General Public License 2.0' in
* Copyright (c) 1997-2014 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 1997-2013 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2017 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2010-2013 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2011-2012 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2013-2014 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2017-2015 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as subject to the "Classpath" *
exception as provided by Oracle in the GPL Version 2 section of the License

Found license 'General Public License 2.0' in *
* Copyright (c) 2011-2012 Oracle and/or its affiliates. All rights reserved. *
The contents of this file are subject to the terms of either the GNU *
* General Public License Version 2 only ("GPL") or the Common Development *
and Distribution License("CDDL") (collectively, the "License"). You *
may not use this file except in compliance with the License. You can *
Oracle designates this particular file as
1.79 libunistring 0.9.9 0ubuntu2
1.79.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.
You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.
TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.
A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains
in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:
a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a
copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.
"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.
8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an
organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a
If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.
@c The GNU General Public License.
@c center Version 3, 29 June 2007

c This file is intended to be included within another document,
c hence no sectioning command or @node.

display
Copyright @copyright{} 2007 Free Software Foundation, Inc. @url{http://fsf.org/}

everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
display

heading Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program---to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.
Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

@heading TERMS AND CONDITIONS

@enumerate 0
@item Definitions.

``This License'' refers to version 3 of the GNU General Public License.

``Copyright'' also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

``The Program'' refers to any copyrightable work licensed under this License. Each licensee is addressed as ``you''. ``Licensees'' and ``recipients'' may be individuals or organizations.

To ``modify'' a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a ``modified version'' of the earlier work or a work ``based on'' the earlier work.

A ``covered work'' means either the unmodified Program or a work based on the Program.

To ``propagate'' a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To ``convey'' a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays ``Appropriate Legal Notices'' to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

@item Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

@item Basic Permissions.
All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

**@item Protecting Users' Legal Rights From Anti-Circumvention Law.**

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

**@item Conveying Verbatim Copies.**

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

@item Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

@enumerate
@item
The work must carry prominent notices stating that you modified it, and giving a relevant date.

@item
The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

@item
You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

@item
If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.
@end enumerate

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.
Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

@enumerate
@item Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
@item Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
@item Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
@item Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A `"User Product"` is either (1) a `"consumer product"`, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, `"normally used"` refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

`"Installation Information"` for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the
recipient, or for the User Product in which it has been modified or
installed. Access to a network may be denied when the modification
itself materially and adversely affects the operation of the network
or violates the rules and protocols for communication across the
network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

@item Additional Terms.

“Additional permissions” are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law. If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own
removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders
of that material) supplement the terms of this License with terms:

@enumerate a
@item
Disclaiming warranty or limiting liability differently from the terms
of sections 15 and 16 of this License; or

@item
Requiring preservation of specified reasonable legal notices or author
attributions in that material or in the Appropriate Legal Notices
displayed by works containing it; or

@item
Prohibiting misrepresentation of the origin of that material, or
requiring that modified versions of such material be marked in
reasonable ways as different from the original version; or
Limiting the use for publicity purposes of names of licensors or authors of the material; or

Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered ``further restrictions'' within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder

Open Source Used In IOx-FogDirector-AC18 1.17.0 806
fails to notify you of the violation by some reasonable means prior to
60 days after the cessation.

Moreover, your license from a particular copyright holder is
reinstated permanently if the copyright holder notifies you of the
violation by some reasonable means, this is the first time you have
received notice of violation of this License (for any work) from that
copyright holder, and you cure the violation prior to 30 days after
your receipt of the notice.

Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under
this License. If your rights have been terminated and not permanently
reinstated, you do not qualify to receive new licenses for the same
material under section 10.

@item Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run
a copy of the Program. Ancillary propagation of a covered work
occurring solely as a consequence of using peer-to-peer transmission
to receive a copy likewise does not require acceptance. However,
nothing other than this License grants you permission to propagate or
modify any covered work. These actions infringe copyright if you do
not accept this License. Therefore, by modifying or propagating a
covered work, you indicate your acceptance of this License to do so.

@item Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically
receives a license from the original licensors, to run, modify and
propagate that work, subject to this License. You are not responsible
for enforcing compliance by third parties with this License.

An ``entity transaction'' is a transaction transferring control of an
organization, or substantially all assets of one, or subdividing an
organization, or merging organizations. If propagation of a covered
work results from an entity transaction, each party to that
transaction who receives a copy of the work also receives whatever
licenses to the work the party's predecessor in interest had or could
give under the previous paragraph, plus a right to possession of the
Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the
rights granted or affirmed under this License. For example, you may
not impose a license fee, royalty, or other charge for exercise of
rights granted under this License, and you may not initiate litigation.
(including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

@item Patents.

A ``contributor'' is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's ``contributor version''.

A contributor's ``essential patent claims'' are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, ``control'' includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a ``patent license'' is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To ``grant'' such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. ``Knowingly relying'' means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a
covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

@item No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

@item Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.
@item Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License ``or any later version'' applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

@item Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM ``AS IS'' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

@item Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

@item Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

@end enumerate

@end enumerate

@heading END OF TERMS AND CONDITIONS

@heading How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the ```copyright``` line and a pointer to where the full notice is found.

@smallexample
@var{one line to give the program's name and a brief idea of what it does.}
Copyright (C) @var{year} @var{name of author}

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see @url{http://www.gnu.org/licenses/}.
@end smallexample

Also add information on how to contact you by electronic and paper mail.
If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

@example
{program} Copyright (C) {year} {name of author}
This program comes with ABSOLUTELY NO WARRANTY; for details type @samp{show w}.
This is free software, and you are welcome to redistribute it under certain conditions; type @samp{show c} for details.
@end example

The hypothetical commands @samp{show w} and @samp{show c} should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an `about box'.

You should also get your employer (if you work as a programmer) or school, if any, to sign a `copyright disclaimer' for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see @url{http://www.gnu.org/licenses/}.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read @url{http://www.gnu.org/philosophy/why-not-lGPL.html}.

The GNU Free Documentation License.
@center Version 1.3, 3 November 2008

@c This file is intended to be included within another document,
@c hence no sectioning command or @node.

@display
@url{http://fsf.org/}
@end display

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

@end display

@enumerate
@item
PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document @dfn{free} in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially.
Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

@item
APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a
section does not fit the above definition of Secondary then it is not
to be designated as Invariant. The Document may contain zero
Invariant Sections. If the Document does not identify any Invariant
Sections then there are none.

The ``Cover Texts'' are certain short passages of text that are listed,
as Front-Cover Texts or Back-Cover Texts, in the notice that says that
the Document is released under this License. A Front-Cover Text may
be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy,
represented in a format whose specification is available to the
general public, that is suitable for revising the document
straightforwardly with generic text editors or (for images composed of
pixels) generic paint programs or (for drawings) some widely available
drawing editor, and that is suitable for input to text formatters or
for automatic translation to a variety of formats suitable for input
to text formatters. A copy made in an otherwise Transparent file
format whose markup, or absence of markup, has been arranged to thwart
or discourage subsequent modification by readers is not Transparent.
An image format is not Transparent if used for any substantial amount
of text. A copy that is not ``Transparent'' is called ``Opaque''.

Examples of suitable formats for Transparent copies include plain
@sc{ascii} without markup, Texinfo input format, La@TeX{} input
format, @acronym{SGML} or @acronym{XML} using a publicly available
@acronym{DTD}, and standard-conforming simple @acronym{HTML},
PostScript or @acronym{PDF} designed for human modification. Examples
of transparent image formats include @acronym{PNG}, @acronym{XCF} and
@acronym{JPG}. Opaque formats include proprietary formats that can be
read and edited only by proprietary word processors, @acronym{SGML} or
@acronym{XML} for which the @acronym{DTD} and/or processing tools are
not generally available, and the machine-generated @acronym{HTML},
PostScript or @acronym{PDF} produced by some word processors for
output purposes only.

The ``Title Page'' means, for a printed book, the title page itself,
plus such following pages as are needed to hold, legibly, the material
this License requires to appear in the title page. For works in
formats which do not have any title page as such, ``Title Page'' means
the text near the most prominent appearance of the work's title,
preceding the beginning of the body of the text.

The ``publisher'' means any person or entity that distributes copies
of the Document to the public.

A section ``Entitled XYZ'' means a named subunit of the Document whose
title either is precisely XYZ or contains XYZ in parentheses following
text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History"). To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

@item
VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

@item
COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent
If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

- State on the Title page the name of the publisher of the Modified Version, as the publisher.
@item
Preserve all the copyright notices of the Document.

@item
Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

@item
Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

@item
Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled ``History'', Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled ``History'' in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the ``History'' section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

@item
For any section Entitled ``Acknowledgements'' or ``Dedications'', Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

@item
Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties---for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.
The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ``History'' in the various original documents, forming one section Entitled ``History''; likewise combine any sections Entitled ``Acknowledgements'', and any sections Entitled ``Dedications''. You must delete all sections Entitled ``Endorsements.''

@item COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

@item AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an ``aggregate'' if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole
aggregate.

ITEM TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

ITEM TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently
reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

@item FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

@item RELICENSING

``Massive Multiauthor Collaboration Site'' (or ``MMC Site'') means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A ``Massive Multiauthor Collaboration'' (or ``MMC'') contained in the site means any set of copyrightable works thus published on the MMC site.

``CC-BY-SA'' means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

``Incorporate'' means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is ``eligible for relicensing'' if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole
or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

@end enumerate

@page
@heading ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

@smallexample
@group
Copyright (C) @var{year} @var{your name}.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled ``GNU Free Documentation License''.
@end group
@end smallexample

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the ``with@dots{}Texts.'' line with this:

@smallexample
@group
with the Invariant Sections being @var{list their titles}, with the Front-Cover Texts being @var{list}, and with the Back-Cover Texts being @var{list}.
@end group
@end smallexample

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.
1.80 quartz 2.3.2

1.80.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written
communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.81 jcl-over-slf4j 1.7.30
1.81.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2004 The Apache Software Foundation.
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/LogFactory.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SLF4JLogFactory.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SLF4JLocationAwareLog.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SLF4JLog.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/NoOpLog.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SimpleLog.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SLF4JLocationAwareLog.java
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/impl/SLF4JLog.java
No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/META-INF/maven/org.slf4j/jcl-over-slf4j/pom.xml
No license file was found, but licenses were detected in source scan.

2004 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1135880174_1613624046.95/0/jcl-over-slf4j-1-7-30-sources-1-jar/org/apache/commons/logging/package.html

1.82 bean-validation-api 2.0.1.Final
1.82.1 Available under license:
Bean Validation API

License: Apache License, Version 2.0
See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.

1.83 jackson-annotations 2.12.1
1.83.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Copyright (c) 2013, 2014 EclipseSource

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.85 jersey-client 2.30.1
1.85.1 Available under license :

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
* by the @authors tag.

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.9.9
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com
JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public
domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 7.2
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology,
Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/Consortium/Legal/
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such
Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient’s receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and
b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient.
No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that
is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise
to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.86 error_prone_annotations 2.3.4
1.86.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/DoNotMock.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/RestrictedApi.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/MustBeClosed.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/FormatMethod.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/FormatString.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/CompatibleWith.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Error Prone Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/RequiredModifiers.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/Var.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/ForOverride.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/concurrent/LazyInit.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/CompileTimeConstant.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/Immutable.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/IncompatibleModifiers.java
* /opt/cola/permits/1177939649_1625570689.35/0/error-prone-annotations-2-3-4-sources-8-
  jar/com/google/errorprone/annotations/SuppressPackageLocation.java

1.87 mapdb 1.0.6

1.87.1 Available under license:
   The Apache Software License, Version 2.0
   http://www.apache.org/licenses/LICENSE-2.0.txt

1.88 p11-kit 0.23.9-2ubuntu0.1

1.88.1 Available under license:
   Redistribution and use in source and binary forms, with or without
   modification, are permitted provided that the following conditions
   are met:

   * Redistributions of source code must retain the above
     copyright notice, this list of conditions and the
     following disclaimer.
   * Redistributions in binary form must reproduce the
     above copyright notice, this list of conditions and
     the following disclaimer in the documentation and/or
     other materials provided with the distribution.
* The names of contributors to this software may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.89 commons-dbcp 1.4

1.89.1 Available under license:

Apache Commons DBCP
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.90 jetty-util 9.4.31.v20200723

1.90.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is
added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution,
if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of
6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient’s rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient’s rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program
Open Source Used In IOx-FogDirector-AC18 1.17.0

(including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=======================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=======================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core
The following artifacts are EPL and ASL 2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache
The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

------
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

------
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.
1.91 kubernetes-model-common 5.4.1

1.91.1 Available under license:

No license file was found, but licenses were detected in source scan.

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2015 Red Hat, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1283431262_1646723909.76/0/kubernetes-model-common-5-4-1-sources-jar/io/fabric8/kubernetes/model/annotation/Generated.java
* /opt/cola/permits/1283431262_1646723909.76/0/kubernetes-model-common-5-4-1-sources-jar/io/fabric8/kubernetes/model/annotation/Kind.java
* /opt/cola/permits/1283431262_1646723909.76/0/kubernetes-model-common-5-4-1-sources-
1.92 selinux 2.7-2build2

1.92.1 Available under license:

This is the Debian packe for libselinux, and it is built from sources obtained from:
http://userspace.selinuxproject.org/trac/wiki/Releases

This package was debianized by Colin Walters <walters@debian.org> on
Thu, 3 Jul 2003 17:10:57 -0400.

This library (libselinux) is public domain software, i.e. not copyrighted.

Warranty Exclusion
-------------------
You agree that this software is a
non-commercially developed program that may contain "bugs" (as that
term is used in the industry) and that it may not function as intended.
The software is licensed "as is". NSA makes no, and hereby expressly
disclaims all, warranties, express, implied, statutory, or otherwise
with respect to the software, including noninfringement and the implied
warranties of merchantability and fitness for a particular purpose.

Limitation of Liability
-----------------------
In no event will NSA be liable for any damages, including loss of data, lost profits, cost of cover, or other special, incidental, consequential, direct or indirect damages arising from the software or the use thereof, however caused and on any theory of liability. This limitation will apply even if NSA has been advised of the possibility of such damage. You acknowledge that this is a reasonable allocation of risk.

However, one file (utils/avcstat.c) is
Copyright: 2004 Red Hat, Inc., James Morris <jmorris@redhat.com>
and is distributed under the terms of the GNU General Public License, version 2.

In addition, The Debian specific package was modified to include an excerpt from the GNU libc package in the file
utils/ia64-inline-syscall.h. The GNU C Library is distributed under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

You should have received a copy of the GNU Lesser General Public License along with the GNU C Library; if not, write to
Free Software Foundation, Inc., 51 Franklin St, Fifth Floor,
Boston, MA 02110-1301, USA.

On Debian systems, the complete text of the GNU Library General Public License can be found in `/usr/share/common-licenses/LGPL-2.1'.

This package is maintained by Manoj Srivastava <srivasta@debian.org>.

The Debian specific changes are 2005, 2006, Manoj Srivastava <srivasta@debian.org>, and distributed under the terms of the GNU General Public License, version 2.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2'.

A copy of the GNU General Public License is also available at
<URL:http://www.gnu.org/copyleft/gpl.html>. You may also obtain it by writing to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

Manoj Srivastava <srivasta@debian.org>
arch-tag: d4250e44-a0e0-4ee0-adb9-2bd7f6eeb27
1.93 libnspr 4.18-1ubuntu1

1.93.1 Available under license:

This package was debianized by Mike Hommey <glandium@debian.org> on Sun, 25 Mar 2007 12:17:27 +0200.

It was downloaded from http://ftp.mozilla.org/pub/mozilla.org/nspr/

Upstream Author: The Mozilla Project.

The NSPR library is licensed under the terms of the Mozilla Public License version 2.0 below.

The original code is copyright (c) 1998-2000 Netscape Communications Corporation.

Mozilla Public License Version 2.0
==================================

1. Definitions
-------------

1.1. "Contributor"

   means each individual or legal entity that creates, contributes to
   the creation of, or owns Covered Software.

1.2. "Contributor Version"

   means the combination of the Contributions of others (if any) used
   by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

   means Covered Software of a particular Contributor.

1.4. "Covered Software"

   means Source Code Form to which the initial Contributor has attached
   the notice in Exhibit A, the Executable Form of such Source Code
   Form, and Modifications of such Source Code Form, in each case
   including portions thereof.

1.5. "Incompatible With Secondary Licenses"

   means

   (a) that the initial Contributor has attached the notice described
       in Exhibit B to the Covered Software; or
(b) that the Covered Software was made available under the terms of
version 1.1 or earlier of the License, but not also under the
terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
Software; or

(b) any new file in Source Code Form that contains any Covered
Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:
(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

______________________________________________________________________________
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or *
*  statutory, including, without limitation, warranties that the      *
*  Covered Software is free of defects, merchantable, fit for a       *
*                                                                      *
particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party’s negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party’s ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.
You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.94 netty-handler 4.1.49.Final

1.94.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/timeout/WriteTimeoutException.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/timeout/IdleStateHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/logging/LogLevel.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/stream/ChunkedFile.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/timeout/TimeoutException.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/timeout/ReadTimeoutHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/logging/LoggingHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/NotSslRecordException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/util/X509KeyManagerWrapper.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/util/SimpleKeyManagerFactory.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/address/DynamicAddressConnectHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/PseudoRandomFunction.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/SslMasterKeyHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/OpenSslPrivateKeyMethod.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/KeyManagerFactoryWrapper.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/TrustManagerFactoryWrapper.java

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/AbstractSniHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/SslCloseCompletionEvent.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/OptionalSslHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/JdkAlpnSslUtils.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/SslCompletionEvent.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/SniCompletionEvent.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/Conscrypt.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ConscryptAlpnSslEngine.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ocsp/OcspClientHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/SslClientHelloHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ocsp/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/SslContextBuilder.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ApplicationProtocolAccessor.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ApplicationProtocolNames.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/OpenSslSessionTicketKey.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/OpenSslJavaxX509Certificate.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ApplicationProtocolNegotiationHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
ar/jar/io/netty/handler/ssl/ClientAuth.java
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/OpenSslKeyMaterialProvider.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/OpenSslKeyMaterial.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/OpenSslCachingKeyMaterialProvider.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/ExtendedOpenSslSession.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
/* No license file was found, but licenses were detected in source scan. */

/* Copyright 2011 The Netty Project */
/* The Netty Project licenses this file to you under the Apache License, */
/* version 2.0 (the "License"); you may not use this file except in compliance */
/* with the License. You may obtain a copy of the License at: */
/* */
/* http://www.apache.org/licenses/LICENSE-2.0 */
/* Unless required by applicable law or agreed to in writing, software */
/* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT */
/* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the */
/* License for the specific language governing permissions and limitations */
/* under the License. */
/* */

Found in path(s):
/* Copyright 2013 The Netty Project */
/* The Netty Project licenses this file to you under the Apache License, */
/* version 2.0 (the "License"); you may not use this file except in compliance */
/* with the License. You may obtain a copy of the License at: */
/* */
/* http://www.apache.org/licenses/LICENSE-2.0 */
/* Unless required by applicable law or agreed to in writing, software */
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/SslHandshakeCompletionEvent.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/traffic/GlobalTrafficShapingHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/traffic/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/traffic/TrafficCounter.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/traffic/ChannelTrafficShapingHandler.java
No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// Try the OpenJDK's proprietary implementation.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/Java8SslUtils.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/PemX509Certificate.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/PemEncoded.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/ReferenceCountedOpenSslClientContext.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/PemPrivateKey.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/OpenSslCertificateException.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/PemValue.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/flow/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/flush/FlushConsolidationHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/DelegatingSslContext.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/ReferenceCountedOpenSslServerContext.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/OpenSslEngine.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/ReferenceCountedOpenSslServerContext.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/OpenSslKeyMaterialManager.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-
  jar/io/netty/handler/ssl/ReferenceCountedOpenSslContext.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2020 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/address/ResolveAddressHandler.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/SslHandshakeTimeoutException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version
* 2.0 (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/flow/FlowControlHandler.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version
* 2.0 (the "License"); you may not use this file except in compliance with the
* License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/SslContext.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/util/FingerprintTrustManagerFactory.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ipfilter/IpFilterRuleType.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/util/InsecureTrustManagerFactory.java
* /opt/ws_local/PERMITS_SQL/1068232349_1594320675.63/0/netty-handler-4-1-49-final-sources-1-jar/io/netty/handler/ssl/SslUtils.java
1.95 ncurses 6.1-1ubuntu1.18.04

1.95.1 Available under license:

Upstream source https://invisible-island.net/ncurses/ncurses-examples.html

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

Files: *

Licence: X11

Files: aclocal.m4 package

Copyright: 2003-2017,2018 by Thomas E. Dickey
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-------------------------------------------------------------------------------

Files: install-sh
Copyright: 1994 X Consortium
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

FSF changes to this file are in the public domain.

Calling this script install-sh is preferred over install.sh, to prevent `make' implicit rules from creating a file called install from it when there is no Makefile.

This script is compatible with the BSD install script, but was written
from scratch. It can only install one file at a time, a restriction shared with many OS's install programs.

On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2`

-- vile: txtmode file-encoding=utf-8
Copyright (c) 1998-2017,2018 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-- vile: txtmode fc=72
-- $Id: COPYING,v 1.6 2018/01/01 12:00:00 tom Exp$
Upstream source https://invisible-island.net/ncurses/ncurses.html
This package is used for testing builds of ncurses.

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

-------------------------------------------------------------------------------
Files: *
Licence: X11

Files: aclocal.m4 package
Copyright: 1996-2017,2018 by Thomas E. Dickey
Licence: X11
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNEC-
TION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not
be used in advertising or otherwise to promote the sale, use or other deal-
ings in this Software without prior written authorization from the X Consor-
tium.

Files: progs/tset.c ncurses/tinfo/read_termcap.c
License: BSD

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS `AS IS` AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

-- vile: txtmode file-encoding=utf-8
Upstream source https://invisible-island.net/ncurses/ncurses-examples.html

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

Files: *
Licence: X11

Files: aclocal.m4 package
Copyright: 2010-2017,2018 by Thomas E. Dickey
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-------------------------------------------------------------------------------
Files: install-sh
Copyright: 1994 X Consortium
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

FSF changes to this file are in the public domain.

Calling this script install-sh is preferred over install.sh, to prevent `make` implicit rules from creating a file called install from it when there is no Makefile.

This script is compatible with the BSD install script, but was written from scratch. It can only install one file at a time, a restriction shared with many OS's install programs.

On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2`

-- vile: txtmode file-encoding=utf-8

1.96 ini4j 0.5.4
1.96.1 Available under license :
    Apache 2
    http://www.apache.org/licenses/LICENSE-2.0.txt

1.97 libidn 2.0.4-1.1ubuntu0.2
1.97.1 Available under license :
    A. Unicode Copyright.

Copyright 1991-2016 Unicode, Inc. All rights reserved.

Certain documents and files on this website contain a legend indicating that "Modification is permitted." Any person is hereby authorized, without fee, to modify such documents and files to create derivative works conforming to the Unicode Standard, subject to Terms and Conditions herein.

Any person is hereby authorized, without fee, to view, use, reproduce, and distribute all documents and files solely for informational purposes and in the creation of products supporting the Unicode Standard, subject to the Terms and Conditions herein.

Further specifications of rights and restrictions pertaining to the use of the particular set of data files known as the "Unicode Character Database" can be found in the License.

Each version of the Unicode Standard has further specifications of rights and restrictions of use. For the book editions (Unicode 5.0 and earlier), these are found on the back of the title page. The online code charts carry specific restrictions. All other files, including online documentation of the core specification for Unicode 6.0 and later, are covered under these general Terms of Use.

No license is granted to "mirror" the Unicode website where a fee is charged for access to the "mirror" site.
Modification is not permitted with respect to this document. All copies of this document must be verbatim.

B. Restricted Rights Legend.
Any technical data or software which is licensed to the United States of America, its agencies and/or instrumentalities under this Agreement is commercial technical data or commercial computer software developed exclusively at private expense as defined in FAR 2.101, or DFARS 252.227-7014 (June 1995), as applicable. For technical data, use, duplication, or disclosure by the Government is subject to restrictions as set forth in DFARS 202.227-7015 Technical Data, Commercial and Items (Nov 1995) and this Agreement. For Software, in accordance with FAR 12-212 or DFARS 227-7202, as applicable, use, duplication or disclosure by the Government is subject to the restrictions set forth in this Agreement.

C. Warranties and Disclaimers.
This publication and/or website may include technical or typographical errors or other inaccuracies. Changes are periodically added to the information herein; these changes will be incorporated in new editions of the publication and/or website. Unicode may make improvements and/or changes in the product(s) and/or program(s) described in this publication and/or website at any time.

If this file has been purchased on magnetic or optical media from Unicode, Inc. the sole and exclusive remedy for any claim will be exchange of the defective media within ninety (90) days of original purchase.

EXCEPT AS PROVIDED IN SECTION C.2, THIS PUBLICATION AND/OR SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. UNICODE AND ITS LICENSORS ASSUME NO RESPONSIBILITY FOR ERRORS OR OMISSIONS IN THIS PUBLICATION AND/OR SOFTWARE OR OTHER DOCUMENTS WHICH ARE REFERENCED BY OR LINKED TO THIS PUBLICATION OR THE UNICODE WEBSITE.

D. Waiver of Damages.
In no event shall Unicode or its licensors be liable for any special, incidental, indirect or consequential damages of any kind, or any damages whatsoever, whether or not Unicode was advised of the possibility of the damage, including, without limitation, those resulting from the following: loss of use, data or profits, in connection with the use, modification or distribution of this information or its derivatives.

E. Trademarks & Logos.
The Unicode Word Mark and the Unicode Logo are trademarks of Unicode, Inc. The Unicode Consortium and Unicode, Inc. are trade names of Unicode, Inc. Use of the information and materials found on this website indicates your acknowledgement of Unicode, Inc.'s exclusive worldwide rights in the Unicode Word Mark, the Unicode Logo, and the Unicode trade names.
The Unicode Consortium Name and Trademark Usage Policy (Trademark Policy) are incorporated herein by reference and you agree to abide by the provisions of the Trademark Policy, which may be changed from time to time in the sole discretion of Unicode, Inc.
All third party trademarks referenced herein are the property of their respective owners.

F. Miscellaneous.
Jurisdiction and Venue. This server is operated from a location in the State of California, United States of America. Unicode makes no representation that the materials are appropriate for use in other locations. If you access this server from other locations, you are responsible for compliance with local laws. This Agreement, all use of this site and any claims and damages resulting from use of this site are governed solely by the laws of the State of California without regard to any principles which would apply the laws of a different jurisdiction. The user agrees that any disputes regarding this site shall be resolved solely in the courts located in Santa Clara County, California.
The user agrees said courts have personal jurisdiction and agree to waive any right to transfer the dispute to any other forum.

Modification by Unicode Unicode shall have the right to modify this Agreement at any time by posting it to this site. The user may not assign any part of this Agreement without Unicodes prior written consent.

Taxes. The user agrees to pay any taxes arising from access to this website or use of the information herein, except for those based on Unicodes net income.

Severability. If any provision of this Agreement is declared invalid or unenforceable, the remaining provisions of this Agreement shall remain in effect.

Entire Agreement. This Agreement constitutes the entire agreement between the parties.

EXHIBIT 1
Unicode Data Files include all data files under the directories
http://www.unicode.org/Public/, http://www.unicode.org/reports/,
http://www.unicode.org/cldr/data/, http://source.icu-project.org/repos/icu/, and
http://www.unicode.org/utility/trac/browser/.

Unicode Data Files do not include PDF online code charts under the directory http://www.unicode.org/Public/.

Software includes any source code published in the Unicode Standard or under the directories
http://www.unicode.org/Public/, http://www.unicode.org/reports/,
http://www.unicode.org/cldr/data/, http://source.icu-project.org/repos/icu/, and
http://www.unicode.org/utility/trac/browser/.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2016 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies of the Data Files or Software, or
(b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
eexercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the “copyright” line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.
A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If
you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

Libidn2 COPYING -- Licensing information.                         *- outline *-
Copyright (C) 2011-2016 Simon Josefsson
See the end for copying conditions.
The source code for the C library (libidn2.a or libidn.so) are licensed under the terms of either the GNU General Public License version 2.0 or later (see the file COPYINGv2) or the GNU Lesser General Public License version 3.0 or later (see the file COPYING.LESSERv3), or both in parallel as here.

The command line tool, self tests, examples, and other auxiliary files, are licensed under the GNU General Public License version 3.0 or later.

The license of the Unicode character data files (which are parsed into static storage in the library) are documented in COPYING.unicode.

Other files are licensed as indicated in each file.

There may be exceptions to these general rules, see each file for precise information.

----------------------------------------------------------------------
This file is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This file is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this file. If not, see <http://www.gnu.org/licenses/>.

1.98 dom4j 2.1.3
1.98.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
 *
 * This software is open source.
 * See the bottom of this file for the licence.
 */
/*
 * Redistribution and use of this software and associated documentation
 * ("Software"), with or without modification, are permitted provided that the
 * following conditions are met:
1. Redistributions of source code must retain copyright statements and
notices. Redistributions must also contain a copy of this document.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

3. The name "DOM4J" must not be used to endorse or promote products derived
from this Software without prior written permission of MetaStuff, Ltd. For
written permission, please contact dom4j-info@metastuff.com.

4. Products derived from this Software may not be called "DOM4J" nor may
"DOM4J" appear in their names without prior written permission of MetaStuff,
Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.

5. Due credit should be given to the DOM4J Project - http://www.dom4j.org

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
*/

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/tree/ElementQNameIterator.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/tree/ElementIterator.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/ElementHandler.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/CDATA.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/Document.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/rule/Action.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/Comment.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/Entity.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-jar/org/dom4j/bean/BeanDocumentFactory.java
*/opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources-
/*
* Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
*
* This software is open source.
* See the bottom of this file for the licence.
*/

/* Redistribution and use of this software and associated documentation
* ("Software"), with or without modification, are permitted provided that the
* following conditions are met:
*
* 1. Redistributions of source code must retain copyright statements and
* notices. Redistributions must also contain a copy of this document.
*
* 2. Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*
* 3. The name "DOM4J" must not be used to endorse or promote products derived
* from this Software without prior written permission of MetaStuff, Ltd. For
* written permission, please contact dom4j-info@metastuff.com.
*
* 4. Products derived from this Software may not be called "DOM4J" nor may
* "DOM4J" appear in their names without prior written permission of MetaStuff,
* Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
*
* 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org
*
* THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND
* ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL METAStuff, LTD. OR ITS CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
* 
* Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources.jar/org/dom4j/DocumentHelper.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources.jar/org/dom4j/io/SAXReader.java
* /opt/ws_local/PERMITS_SQL/1034543185_1588830303.8/0/dom4j-2-1-3-sources.jar/org/dom4j/io/SAXHelper.java

### 1.99 commons-digester 2.1

#### 1.99.1 Available under license:

Apache Commons Digester
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

 TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.100 zlib 1.2.11.dfsg-0ubuntu2

1.100.1 Available under license:

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be
1.101 jradius-dictionary 1.1.6-SNAPSHOT

1.101.1 Available under license:

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you
can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.
In addition, mere aggregation of another work not based on the Program with the
Program (or with a work based on the Program) on a volume of a storage or
distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under
Section 2) in object code or executable form under the terms of Sections 1 and
2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source
code, which must be distributed under the terms of Sections 1 and 2 above
on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to
give any third party, for a charge no more than your cost of physically
performing source distribution, a complete machine-readable copy of the
corresponding source code, to be distributed under the terms of Sections 1
and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed only
for noncommercial distribution and only if you received the program in
object code or executable form with such an offer, in accord with
Subsection b above.)

The source code for a work means the preferred form of the work for making
modifications to it. For an executable work, complete source code means all
the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation
of the executable. However, as a special exception, the source code
distributed need not include anything that is normally distributed (in either
source or binary form) with the major components (compiler, kernel, and so on)
of the operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source
code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as
expressly provided under this License. Any attempt otherwise to copy, modify,
sublicense or distribute the Program is void, and will automatically terminate
your rights under this License. However, parties who have received copies, or
rights, from you under this License will not have their licenses terminated so
long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it.
However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In
such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.
signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL, but only where Sun has expressly included in the particular source file’s header the words "Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the LICENSE file that accompanied this code.”

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

==========================================================================

Coova JRadius License:

The Coova extensions to JRadius library are licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within in the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/. See source files for details regarding stand-alone applications contained herein which are released under the GPL.

Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.
Original JRadius License:

The JRadius core library is licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/

This software is OSI Certified Open Source Software. OSI Certified is a certification mark of the Open Source Initiative.

Stand-alone JRadius Applications:

* JRadiusSimulator
* JRadiusWiFiClient
* RadClient
* RadBench
* RadiusDictionary

Copyright 2004-2006 PicoPoint, B.V.
Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

==========================================================================

Coova JRadius License:

The Coova JRadius core libraries are licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://www.coova.org/JRadius. See source files for details regarding stand-alone applications contained herein which are released under the GPL.

Copyright 2006-2010 Coova Technologies, LLC. All rights reserved.

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

==========================================================================

Original JRadius License:

The JRadius core library is licensed under the The GNU Library or "Lesser"
General Public License (LGPL), while stand-alone applications within the JRADIUS project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/

This software is OSI Certified Open Source Software. OSI Certified is a certification mark of the Open Source Initiative.

Stand-alone JRADIUS Applications:

- JRADIUSimulator
- JRADIUSWiFiClient
- RadClient
- RadBench
- RadiusDictionary

Copyright 2004-2006 PicoPoint, B.V.
Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

==========================================================================

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

(This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.)

Preamble

The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it.  You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.
When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating
system.

Although the Lesser General Public License is Less protective of the
users' freedom, it does ensure that the user of a program that is
linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, whereas the latter must
be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other
program which contains a notice placed by the copyright holder or
other authorized party saying it may be distributed under the terms of
this Lesser General Public License (also called "this License").
Each licensee is addressed as "you".
A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above,

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

{description}
Copyright (C) {year} {fullname}

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

{signature of Ty Coon}, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The GNU General Public License (GPL)
Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to
share and change it. By contrast, the GNU General Public License is intended
to guarantee your freedom to share and change free software--to make sure
the software is free for all its users. This General Public License applies
to most of the Free Software Foundation's software and to any other program
whose authors commit to using it. (Some other Free Software Foundation
software is covered by the GNU Library General Public License instead.) You
can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our
General Public Licenses are designed to make sure that you have the freedom
to distribute copies of free software (and charge for this service if you
wish), that you receive source code or can get it if you want it, that you
can change the software or use pieces of it in new free programs; and that
you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to
deny you these rights or to ask you to surrender the rights. These
restrictions translate to certain responsibilities for you if you distribute
copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or
for a fee, you must give the recipients all the rights that you have. You
must make sure that they, too, receive or can get the source code. And you
must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2)
offer you this license which gives you legal permission to copy, distribute
and/or modify the software.

Also, for each author's protection and ours, we want to make certain that
everyone understands that there is no warranty for this free software. If
the software is modified by someone else and passed on, we want its
recipients to know that what they have is not the original, so that any
problems introduced by others will not reflect on the original authors'
reputations.

Finally, any free program is threatened constantly by software patents. We
wish to avoid the danger that redistributors of a free program will
individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with
Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and
any other pertinent obligations, then as a consequence you may not
distribute the Program at all. For example, if a patent license would not
permit royalty-free redistribution of the Program by all those who
receive copies directly or indirectly through you, then the only way you
could satisfy both it and this License would be to refrain entirely from
distribution of the Program.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply and
the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents
or other property right claims or to contest validity of any such claims;
this section has the sole purpose of protecting the integrity of the free
software distribution system, which is implemented by public license
practices. Many people have made generous contributions to the wide range of
software distributed through that system in reliance on consistent
application of that system; it is up to the author/donor to decide if he or
she is willing to distribute software through any other system and a
licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a
consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain
countries either by patents or by copyrighted interfaces, the original
copyright holder who places the Program under this License may add an
explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of
the General Public License from time to time. Such new versions will be
similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free programs
whose distribution conditions are different, write to the author to ask
for permission. For software which is copyrighted by the Free Software
Foundation, write to the Free Software Foundation; we sometimes make
exceptions for this. Our decision will be guided by the two goals of
preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR
THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible
use to the public, the best way to achieve this is to make it free software
which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey the
exclusion of warranty; and each file should have at least the "copyright"
line and a pointer to where the full notice is found.

One line to give the program’s name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by the
Free Software Foundation; either version 2 of the License, or (at your
option) any later version.
This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when
it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision
comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is
free software, and you are welcome to redistribute it under certain
conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may be
called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General Public
License instead of this License.
GNU Lesser General Public License

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc. 59 Temple Place,
Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and
distribute verbatim copies of this license document, but changing it is
not allowed.
[This is the first released version of the Lesser GPL. It also counts as
the successor of the GNU Library Public License, version 2, hence the
version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to
share and change it. By contrast, the GNU General Public Licenses are
intended to guarantee your freedom to share and change free software--to
make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially
designated software packages--typically libraries--of the Free Software
Foundation and other authors who decide to use it. You can use it too, but
we suggest you first think carefully about whether this license or the
ordinary General Public License is the better strategy to use in any
particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not
price. Our General Public Licenses are designed to make sure that you have
the freedom to distribute copies of free software (and charge for this
service if you wish); that you receive source code or can get it if you want
it; that you can change the software and use pieces of it in new free
programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for you if
you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for
a fee, you must give the recipients all the rights that we gave you. You
must make sure that they, too, receive or can get the source code. If you
link other code with the library, you must provide complete object files to
the recipients, so that they can relink them with the library after making
changes to the library and recompiling it. And you must show them these
terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library,
and (2) we offer you this license, which gives you legal permission to copy,
distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no
warranty for the free library. Also, if the library is modified by someone
else and passed on, the recipients should know that what they have is not
the original version, so that the original author's reputation will not be
affected by problems that might be introduced by others.
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run. TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure
layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of
this License to do so, and all its terms and conditions for copying, 
distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the 
Library), the recipient automatically receives a license from the original 
licensor to copy, distribute, link with or modify the Library subject to 
these terms and conditions. You may not impose any further restrictions on 
the recipients’ exercise of the rights granted herein. You are not 
responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent 
infringement or for any other reason (not limited to patent issues), 
conditions are imposed on you (whether by court order, agreement or 
otherwise) that contradict the conditions of this License, they do not 
excuse you from the conditions of this License. If you cannot distribute so 
as to satisfy simultaneously your obligations under this License and any 
other pertinent obligations, then as a consequence you may not distribute 
the Library at all. For example, if a patent license would not permit 
royalty-free redistribution of the Library by all those who receive copies 
directly or indirectly through you, then the only way you could satisfy both 
it and this License would be to refrain entirely from distribution of the 
Library.

If any portion of this section is held invalid or unenforceable under any 
perticular circumstance, the balance of the section is intended to apply, 
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents 
or other property right claims or to contest validity of any such claims; 
this section has the sole purpose of protecting the integrity of the free 
software distribution system which is implemented by public license 
practices. Many people have made generous contributions to the wide range of 
software distributed through that system in reliance on consistent 
application of that system; it is up to the author/donor to decide if he or 
she is willing to distribute software through any other system and a 
licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a 
consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain 
countries either by patents or by copyrighted interfaces, the original 
copyright holder who places the Library under this License may add an 
explicit geographical distribution limitation excluding those countries, so 
that distribution is permitted only in or among countries not thus 
excluded. In such case, this License incorporates the limitation as if 
written in the body of this License.
13. The Free Software Foundation may publish revised and/or new versions of
the Lesser General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Library does not specify a license version
number, you may choose any version ever published by the Free Software
Foundation.

14. If you wish to incorporate parts of the Library into other free programs
whose distribution conditions are incompatible with these, write to the
author to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of
preserving the free status of all derivatives of our free software and of
promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR
THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO
THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY
PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR
CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO
LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR
THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER
SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES. END OF TERMS AND CONDITIONS How to Apply These
Terms to Your New Libraries If you develop a new library, and you want it to
be of the greatest possible use to the public, we recommend making it free
software that everyone can redistribute and change. You can do so by
permitting redistribution under these terms (or, alternatively, under the
terms of the ordinary General Public License).
To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and an idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

signature of Ty Coon, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.102 libffi 3.2.1 8
1.102.1 Available under license:

libffi - Copyright (c) 1996-2014 Anthony Green, Red Hat, Inc and others. See source files for details.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.103 shiro 1.7.1

1.103.1 Available under license :

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
**1.104.1 Available under license:**

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

---

**Eclipse Public License - v 1.0**

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly
perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.
When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN
"AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to
time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and

* The Eclipse Public 1.0 License
Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html
org.ow2.asm:asm-commons
org.ow2.asm:asm

-----
Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----
MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by
Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.105 librbd 12.2.13-0ubuntu0.18.04.10

1.105.1 Available under license:
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: ceph
Upstream-Contact: Sage Weil <sage@newdream.net>
Source: http://ceph.com/
Files-Excluded:
  debian
  src/civetweb/examples/websocket_client/ssl/server.key.orig
  src/civetweb/resources/cert/client.key.orig
  src/civetweb/resources/cert/server.key.orig
  src/erasure-code/jerasure/jerasure/Examples/makefile.orig
  src/erasure-code/jerasure/jerasure/include/config.h.in-
  src/erasure-code/jerasure/jerasure/makefile.orig

Files: *
Copyright: 2004-2014 Sage Weil <sage@newdream.net>
  2004-2014 Inktank <info@inktank.com>
    Inktank, Inc
    Inktank Storage, Inc.
  2012-2014 Red Hat <contact@redhat.com>
  2013-2014 Cloudwatt <licre.licensing@cloudwatt.com>
  2013 CohortFS, LLC
  2004-2011 Dreamhost
  2013 eNovance SAS <licensing@enovance.com>
  2014 Adam Crume <adamcrume@gmail.com>
  2012 Florian Haas, hastexo
  2010 Greg Farnum <gregf@hq.newdream.net>
  2014 John Spray <john.spray@inktank.com
  2004-2012 New Dream Network
  2014 Sebastien Ponce <sebastien.ponce@cern.ch>
  2011 Stanislav Sedov <stas@FreeBSD.org>
  2013-2014 UnitedStack <haomai@unitedstack.com>
  2011 Wido den Hollander <wido@widodh.nl>
License: LGPL-2.1

Files: src/erasure-code/jerasure/ErasureCode*
  src/erasure-code/ErasureCode*
  src/erasure-code/isa/*
  src/include/str_map.h
src/test/common/test_str_map.cc
src/test/erasure-code/*
src/test/rgw/test_rgw_manifest.cc

Copyright: 2014 CERN/Switzerland
2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
2014 Red Hat <contact@redhat.com>
2013 eNovance SAS <licensing@enovance.com>

License: LGPL-2.1+

Files: src/erasure-code/isa/isa-l/erasure_code/*
Copyright: 2011-2014 Intel Corporation
License: BSD-3-clause

Comment:
Additional Grant of Patent Rights

Software means the rocksdb software distributed by Facebook, Inc.

Facebook hereby grants you a perpetual, worldwide, royalty-free, non-exclusive, irrevocable (subject to the termination provision below) license under any rights in any patent claims owned by Facebook, to make, have made, use, sell, offer to sell, import, and otherwise transfer the Software. For avoidance of doubt, no license is granted under Facebooks rights in any patent claims that are infringed by (i) modifications to the Software made by you or a third party, or (ii) the Software in combination with any software or other technology provided by you or a third party.

The license granted hereunder will terminate, automatically and without notice, for anyone that makes any claim (including by filing any lawsuit, assertion or other action) alleging (a) direct, indirect, or contributory infringement or inducement to infringe any patent: (i) by Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, (ii) by any party if such claim arises in whole or in part from any software, product or service of Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, or (iii) by any party relating to the Software; or (b) that any right in any patent claim of Facebook is invalid or unenforceable.

Files: src/rocksdb/util/xxhash.*
Copyright: 2012-2014, Yann Collet.
License: BSD-2-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: src/mount/canonicalize.c  
   src/test/common/test_config.cc  
   src/test/crush/TestCrushWrapper.cc  
   src/test/common/Throttle.cc  
   src/test/objectstore/chain_xattr.cc  
   src/test/mon/mon-test-helpers.sh  
   src/test/objectstore/chain_xattr.cc  
   src/test/osd/osd-test-helpers.sh  
   src/ceph-disk  
   src/stop.sh

Copyright: 1993      Rick Sladkey <jrs@world.std.com>  
   2013      Inktank <info@inktank.com>  
   2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
License: LGPL-2+

Files: src/os/btrfs_ioctl.h  
   src/test/mon/PGMap.cc

Copyright: 2007 Oracle. All rights reserved.  
   2014 Inktank <info@inktank.com>
License: GPL-2

Files: src/common/ceph_hash.cc
License: public-domain
This file uses Robert Jenkin's hash function as detailed at:
   
   http://burtleburtle.net/bob/hash/evahash.html
This is in the public domain.

Files: src/common/bloom_filter.hpp
Copyright: 2000 Arash Partow
License: Boost-Software-License-1.0
Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Files: src/common/crc32c_intel*
Copyright: 2012-2013 Intel Corporation All Rights Reserved.
License: BSD-3-clause

Files: src/common/sctp_crc32.c
Copyright: 2001-2007, by Cisco Systems, Inc. All rights reserved,
2004-2006 Intel Corporation - All Rights Reserved
License: BSD-3-clause

Files: src/erasure-code/jerasure/gf-complete/*/*
Copyright: 2013 James S. Plank
Ethan L. Miller
Kevin M. Greenan
Benjamin A. Arnold
John A. Burnum
Adam W. Disney
Allen C. McBride
License: BSD-3-clause
Comment:
https://bitbucket.org/jimplank/gf-complete
Files: src/erasure-code/jerasure/jerasure/*/*
Copyright: 2011-2013 James S. Plank <plank@cs.utk.edu>
2013 Kevin Greenan
License: BSD-3-clause

Files: src/gtest/*
Copyright: 2008, Google Inc.
License: BSD-3-clause

Files: src/civetweb/*
Copyright: 2004-2013 Sergey Lyubka
2013-2014 the Civetweb developers
License: Expat

Files: src/json_spirit/*
Copyright: 2007-2011, John W. Wilkinson
License: Expat

Files: src/java/native/ScopedLocalRef.h
   src/java/native/JniConstants.*
Copyright: 2010 The Android Open Source Project
License: Apache-2.0
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The complete text of the Apache License, Version 2.0
can be found in "/usr/share/common-licenses/Apache-2.0".

Files: src/libs3/*
Copyright: 2008 Bryan Ischo <bryan@ischo.com>
License: GPL-3/OpenSSL
libs3 is free software: you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software
Foundation, version 3 of the License.

In addition, as a special exception, the copyright holders give
permission to link the code of this library and its programs with the
OpenSSL library, and distribute linked combinations including the two.
libs3 is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE. See the GNU General Public License for more
details.

The complete text of the GNU General Public License version 3
can be found in `/usr/share/common-licenses/GPL-3' file.

Files: src/mount/mtab.c
Copyright: util-linux-ng AUTHORS
License: GPL-2+
Comment:
"mount/fstab.c" from line 559:
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/mount-deprecated/fstab.c?h=v2.22#n559
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/README.licensing

Files: src/test/librbd/fsx.c
Copyright: 1991, NeXT Computer, Inc.
License: APSL-2.0
The contents of this file constitute Original Code as defined in and
are subject to the Apple Public Source License Version 2.0 (the
"License"). You may not use this file except in compliance with the
License. Please obtain a copy of the License at

This Original Code and all software distributed under the License are
distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. Please see the
License for the specific language governing rights and limitations
under the License.
Comment:
http://codemonkey.org.uk/projects/fsx/
http://codemonkey.org.uk/projects/fsx/fsx-macosforge/fsx.c

Files: man/*
   debian/man/*
Copyright: 2010-2014, Inktank Storage, Inc. and contributors.
License: CC-BY-SA-3.0

Files: debian/*
Copyright: 2010 Sage Weil <sage@newdream.net>
  2010 Canonical, Ltd.
  2011-2013 Lszl Bszrmnyi (GCS) <gcs@debian.org>
  2013-2014 James Page <james.page@ubuntu.com>
  2014 Dmitry Smirnov <onlyjob@debian.org>
License: LGPL-2.1

License: GPL-2
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2` file.

License: GPL-2+
This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2`.

License: LGPL-2.1
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 2.1 as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

License: LGPL-2.1+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 2 (or later) as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2`.

License: Expat
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Comment:
This license also known as "MIT" however FSF consider "MIT" labelling ambiguous and copyright-format specification recommend to label such license as "Expat".

License: CC-BY-SA-3.0
Creative Commons Attribution-ShareAlike 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not
be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.

c. "Creative Commons Compatible License" means a license that is listed at http://creativecommons.org/compatiblelicenses that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.

d. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

e. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.

f. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

g. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
h. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

i. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

j. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

k. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive,
perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:
a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.

b. You may Distribute or Publicly Perform an Adaptation only under the terms of: (i) this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-ShareAlike 3.0 US); (iv) a Creative Commons Compatible License. If you license the Adaptation under one of the licenses mentioned in (iv), you must comply with the terms of that license. If you license the Adaptation under the terms of any of the licenses mentioned in (i), (ii) or (iii) (the "Applicable License"), you must comply with the terms of the Applicable License generally and the following provisions: (I) You must include a copy of, or the URI for, the Applicable License with every copy of each Adaptation You Distribute or Publicly Perform; (II) You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License; (III) You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform; (IV) when You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection
apart from the Adaptation itself to be made subject to the terms of the
Applicable License.

c. If You Distribute, or Publicly Perform the Work or any Adaptations or
Collections, You must, unless a request has been made pursuant to
Section 4(a), keep intact all copyright notices for the Work and
provide, reasonable to the medium or means You are utilizing: (i) the
name of the Original Author (or pseudonym, if applicable) if supplied,
and/or if the Original Author and/or Licensor designate another party or
parties (e.g., a sponsor institute, publishing entity, journal) for
attribution ("Attribution Parties") in Licensor's copyright notice,
terms of service or by other reasonable means, the name of such party or
parties; (ii) the title of the Work if supplied; (iii) to the extent
reasonably practicable, the URI, if any, that Licensor specifies to be
associated with the Work, unless such URI does not refer to the
copyright notice or licensing information for the Work; and (iv)
consistent with Section 3(b), in the case of an Adaptation, a credit
identifying the use of the Work in the Adaptation (e.g., "French
translation of the Work by Original Author," or "Screenplay based on
original Work by Original Author"). The credit required by this Section
4(c) may be implemented in any reasonable manner; provided, however,
that in the case of a Adaptation or Collection, at a minimum such credit
will appear, if a credit for all contributing authors of the Adaptation
or Collection appears, then as part of these credits and in a manner at
least as prominent as the credits for the other contributing authors.
For the avoidance of doubt, You may only use the credit required by this
Section for the purpose of attribution in the manner set out above and,
by exercising Your rights under this License, You may not implicitly or
explicitly assert or imply any connection with, sponsorship or
endorsement by the Original Author, Licensor and/or Attribution Parties,
as appropriate, of You or Your use of the Work, without the separate,
express prior written permission of the Original Author, Licensor and/or
Attribution Parties.

d. Except as otherwise agreed in writing by the Licensor or as may be
otherwise permitted by applicable law, if You Reproduce, Distribute or
Publicly Perform the Work either by itself or as part of any Adaptations
or Collections, You must not distort, mutilate, modify or take other
derogatory action in relation to the Work which would be prejudicial to
the Original Author's honor or reputation. Licensor agrees that in those
jurisdictions (e.g. Japan), in which any exercise of the right granted
in Section 3(b) of this License (the right to make Adaptations) would be
deemed to be a distortion, mutilation, modification or other derogatory
action prejudicial to the Original Author's honor and reputation, the
Licensor will waive or not assert, as appropriate, this Section, to the
fullest extent permitted by the applicable national law, to enable You
to reasonably exercise Your right under Section 3(b) of this License
(right to make Adaptations) but not otherwise.
5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use
by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License.

Creative Commons may be contacted at http://creativecommons.org/.

License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.106 open-ldap 2.4.45+dfsg-1ubuntu1.10
1.106.1 Available under license:
Copyright 1998-2017 The OpenLDAP Foundation. All rights reserved.

COPYING RESTRICTIONS APPLY.

See COPYRIGHT and LICENSE files in the top-level directory of this distribution (i.e., ../../../COPYRIGHT and ../../../LICENSE, respectively).

/*
 * Copyright (C) 2000 Pierangelo Masarati, <ando@sys-net.it>
 * All rights reserved.
 *
 * Permission is granted to anyone to use this software for any purpose
 * on any computer system, and to alter it and redistribute it, subject
 * to the following restrictions:
 *
 * 1. The author is not responsible for the consequences of use of this
 *    software, no matter how awful, even if they arise from flaws in it.
 *
 * 2. The origin of this software must not be misrepresented, either by
 *    explicit claim or by omission. Since few users ever read sources,
 *    credits should appear in the documentation.
 *
 * 3. Altered versions must be plainly marked as such, and must not be
 *    misrepresented as being the original software. Since few users
 *    ever read sources, credits should appear in the documentation.
 *
 * 4. This notice may not be removed or altered.
 */

Copyright 1998-2017 The OpenLDAP Foundation. All rights reserved.

COPYING RESTRICTIONS APPLY.

See COPYRIGHT and LICENSE files in the top-level directory of this distribution (i.e., ../../../COPYRIGHT and ../../../LICENSE, respectively).

---


Copyright (c) 1998-1999 NeoSoft, Inc.
All Rights Reserved.

This software may be used, modified, copied, distributed, and sold, in both source and binary form provided that these copyrights are retained and their terms are followed.

Under no circumstances are the authors or NeoSoft Inc. responsible
for the proper functioning of this software, nor do the authors
assume any liability for damages incurred with its use.

Redistribution and use in source and binary forms are permitted
provided that this notice is preserved and that due credit is given
to NeoSoft, Inc.

NeoSoft, Inc. may not be used to endorse or promote products derived
from this software without specific prior written permission. This
software is provided `as is` without express or implied warranty.

Requests for permission may be sent to NeoSoft Inc, 1770 St. James Place,
Suite 500, Houston, TX, 77056.
Copyright 2011-2017 Howard Chu, Symas Corp.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted only as authorized by the OpenLDAP
Public License.

A copy of this license is available in the file LICENSE in the
top-level directory of the distribution or, alternatively, at

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Individual files and/or contributed packages may be copyright by
other parties and/or subject to additional restrictions.

This work also contains materials derived from public sources.

Additional information about OpenLDAP can be obtained at
Copyright 1998-2017 The OpenLDAP Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted only as authorized by the OpenLDAP
Public License.

A copy of this license is available in the file LICENSE in the
top-level directory of the distribution or, alternatively, at

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Individual files and/or contributed packages may be copyright by
other parties and/or subject to additional restrictions.
This work is derived from the University of Michigan LDAP v3.3 distribution. Information concerning this software is available at <http://www.umich.edu/~dirsvcs/ldap/ldap.html>.

This work also contains materials derived from public sources.

Additional information about OpenLDAP can be obtained at <http://www.openldap.org/>.

---

Portions Copyright 2001-2006 IBM Corporation.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted only as authorized by the OpenLDAP Public License.

---

Portions Copyright 1999-2008 Symas Corporation.
Portions Copyright 2007-2011 Gavin Henry.
Portions Copyright 2007-2011 Suretec Systems Ltd.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that this notice is preserved. The names of the copyright holders may not be used to endorse or promote products derived from this software without their specific prior written permission. This software is provided ``as is'' without express or implied warranty.

---

Portions Copyright (c) 1992-1996 Regents of the University of Michigan.
All rights reserved.

Redistribution and use in source and binary forms are permitted provided that this notice is preserved and that due credit is given to the University of Michigan at Ann Arbor. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. This software is provided ``as is'' without express or implied warranty.
The OpenLDAP Public License
Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation
("Software"), with or without modification, are permitted provided
that the following conditions are met:

1. Redistributions in source form must retain copyright statements
   and notices,

2. Redistributions in binary form must reproduce applicable copyright
   statements and notices, this list of conditions, and the following
   disclaimer in the documentation and/or other materials provided
   with the distribution, and

3. Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time.
Each revision is distinguished by a version number. You may use
this Software under terms of this license revision or under the
terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS
CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT
SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S)
OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in
advertising or otherwise to promote the sale, use or other dealing
in this Software without specific, written prior permission. Title
to copyright in this Software shall at all times remain with copyright
holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City,
California, USA. All Rights Reserved. Permission to copy and
distribute verbatim copies of this document is granted.
1.107 hibernate 5.2.17.Final

1.107.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU
General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the
software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically
libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest
you first think carefully about whether this license or the ordinary General Public License is the better strategy to
use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are
designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if
you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces
of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to
surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the
rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code
with the library, you must provide complete object files to the recipients, so that they can relink them with the
library after making changes to the library and recompiling it. And you must show them these terms so they know
their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which
gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the
library is modified by someone else and passed on, the recipients should know that what they have is not the original
version, so that the original author's reputation will not be affected by problems that might be introduced by others.
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

“Source code” for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.
In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on
the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given
copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the
ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not
make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public
License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or
executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete
corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above
on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent
access to copy the source code from the same place satisfies the requirement to distribute the source code, even
though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by
being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a
derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the
Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is
therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for
the work may be a derivative work of the Library even though the source code is not. Whether this is true is
especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold
for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and
small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether
it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall
under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms
of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly
with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the
Library to produce a work containing portions of the Library, and distribute that work under terms of your choice,
provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy. For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you
permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. You may not distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting
the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

1.108 libpwquality 1.4.0-2

1.108.1 Available under license:

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Files: *
Copyright: Copyright (c) 1996 Cristian Gafton <gafton@redhat.com>
          Copyright (c) 2011 Red Hat, Inc
          Copyright (c) 2011 Tomas Mraz <tm@t8m.info>
          Copyright (c) Philip W. Dalrymple III <pwd@mdtsoft.com>
License: libpwquality or GPL-2+

Files: debian/*
Copyright: Copyright (c) 2012 Canonical Ltd
License: GPL-3+

License: libpwquality
Redistribution and use in source and binary forms of libpwquality, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain any existing copyright notice, and this entire permission notice in its entirety,
including the disclaimer of warranties.

2. Redistributions in binary form must reproduce all prior and current copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of any author may not be used to endorse or promote products derived from this software without their specific prior written permission.

License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
Comment: You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

On Debian systems the full text of the GNU General Public License can be found in the `/usr/share/common-licenses/GPL-2` file.

License: GPL-3+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
Comment: You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-3`.

1.109 iconv 2.27
1.109.1 Available under license: GNU GENERAL PUBLIC LICENSE

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers’ and authors’ protection, the GPL clearly explains that there is no warranty for this free software. For both users’ and authors’ sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying,
distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.
Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of
it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently
reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted
by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying
the work, and under which the third party grants, to any of the
parties who would receive the covered work from you, a discriminatory
patent license (a) in connection with copies of the covered work
conveyed by you (or copies made from those copies), or (b) primarily
for and in connection with specific products or compilations that
contain the covered work, unless you entered into that arrangement,
or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting
any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot convey a
covered work so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may
not convey it at all. For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

d) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the source code distributed need not include anything that is normally
distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the Free
Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
MA 02110-1301, USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library "Frob" (a library for tweaking knobs) written by James Random Hacker.
That's all there is to it!

1.110 jakarta-ws-rs-api 2.1.6

1.110.1 Available under license:

# Notices for the Jakarta RESTful Web Services Project

This content is produced and maintained by the **Jakarta RESTful Web Services** project.

* Project home: https://projects.eclipse.org/projects/ee4j.jaxrs

## Trademarks

**Jakarta RESTful Web Services** is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jaxrs-api

## Third-party Content

This project leverages the following third party content.
javaee-api (7.0)

* License: Apache-2.0 AND W3C

JUnit (4.11)

* License: Common Public License 1.0

Mockito (2.16.0)

* Project: http://site.mockito.org
* Source: https://github.com/mockito/mockito/releases/tag/v2.16.0

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the
contents of the Program, including, for purposes of clarity any new file
in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely
in each case in order to link to, bind by name, or subclass the Program
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available
in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making
modifications, including but not limited to software source code,
documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,
Version 2.0, or any later versions of that license, including any
exceptions or additional permissions as identified by the initial
Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby
grants Recipient a non-exclusive, worldwide, royalty-free copyright
license to reproduce, prepare Derivative Works of, publicly display,
publicly perform, Distribute and sublicense the Contribution of such
Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby
grants Recipient a non-exclusive, worldwide, royalty-free patent
license under Licensed Patents to make, use, sell, offer to sell,
import and otherwise transfer the Contribution of such Contributor,
if any, in Source Code or other form. This patent license shall
apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

    a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

    b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

        i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness
for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all
liability for damages, including direct, indirect, special,
incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients’ rights
in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any
party to be under a license that satisfies the requirements
of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the
Program (i) is combined with other material in a separate file or
files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in
Exhibit A of this Agreement, then the Program may be made available
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability (“notices”) contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product
offering should do so in a manner which does not create potential
liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
(“Commercial Contributor”) hereby agrees to defend and indemnify every
other Contributor (“Indemnified Contributor”) against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program
in a commercial product offering. The obligations in this section do not
apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),
Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

# The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis
or for a fee, you must give the recipients all the rights that you have.
You must make sure that they, too, receive or can get the source code.
And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software patents.
We wish to avoid the danger that redistributors of a free program will
individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must
be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a
notice placed by the copyright holder saying it may be distributed under
the terms of this General Public License. The "Program", below, refers
to any such program or work, and a "work based on the Program" means
either the Program or any derivative work under copyright law: that is
to say, a work containing the Program or a portion of it, either
verbatim or with modifications and/or translated into another language.
(Hereinafter, translation is included without limitation in the term
"modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of running
the Program is not restricted, and the output from the Program is
covered only if its contents constitute a work based on the Program
(independent of having been made by running the Program). Whether that
is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source
code as you receive it, in any medium, provided that you conspicuously
and appropriately publish on each copy an appropriate copyright notice
and disclaimer of warranty; keep intact all the notices that refer to
this License and to the absence of any warranty; and give any other
recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if
Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

---

1.111 kubernetes-model-node 5.4.1

1.111.1 Available under license:
No license file was found, but licenses were detected in source scan.

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

* Copyright (C) 2015 Red Hat, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and limitations under the License.
*

Found in path(s):
* /opt/cola/permits/1283432457_1646721441.48/0/kubernetes-model-node-5-4-1-sources-jar/manifest.vm

1.112 jakarta-inject 2.6.1
1.112.1 Available under license:

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the
contents of the Program, including, for purposes of clarity any new file
in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely
in each case in order to link to, bind by name, or subclass the Program
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available
in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making
modifications, including but not limited to software source code,
documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,
Version 2.0, or any later versions of that license, including any
exceptions or additional permissions as identified by the initial
2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby
grants Recipient a non-exclusive, worldwide, royalty-free copyright
license to reproduce, prepare Derivative Works of, publicly display,
publicly perform, Distribute and sublicense the Contribution of such
Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby
grants Recipient a non-exclusive, worldwide, royalty-free patent
license under Licensed Patents to make, use, sell, offer to sell,
import and otherwise transfer the Contribution of such Contributor,
if any, in Source Code or other form. This patent license shall
apply to the combination of the Contribution and the Program if, at
the time the Contribution is added by the Contributor, such addition
of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other
combinations which include the Contribution. No hardware per se is
licensed hereunder.

c) Recipient understands that although each Contributor grants the
licenses to its Contributions set forth herein, no assurances are
provided by any Contributor that the Program does not infringe the
patent or other intellectual property rights of any other entity.
Each Contributor disclaims any liability to Recipient for claims
brought by any other entity based on infringement of intellectual
property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby
assumes sole responsibility to secure any other intellectual
property rights needed, if any. For example, if a third party
patent license is required to allow Recipient to Distribute the
Program, it is Recipient's responsibility to acquire that license
before distributing the Program.

d) Each Contributor represents that to its knowledge it has
sufficient copyright rights in its Contribution, if any, to grant
the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no
Contributor makes additional grants to any Recipient (other than
those set forth in this Agreement) as a result of such Recipient's
receipt of the Program under the terms of a Secondary License
(if permitted under the terms of Section 3).

3. REQUIREMENTS
3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients’ rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (“notices”) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its...
Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.)

The GNU General Public License is designed to make sure that free software is maintained free. Software that changes, adds or removes features and/or distributions the source code must do so freely and without restriction.

The most basic goal of this License is to secure the right for free software to change in the same way the Internet itself does. When there exist an effective way to do this, it must always be possible for any individual to possess and change the corresponding source code. The purpose of this section is to outline the general principles of the license, and to lay down some terms and conditions for preserving the free status of future releases. GNU versions of the Free Software Foundation's programs accept minor exceptions, but the GNU GPL Version 2 clearly does not.

Both as an嗜酸菌ist and as a programmer, I want GNU software to be free software that anyone can modify and share. I believe that users have the right to changing a program in order to make it work in their own way and that further, in spirit if not in letter, this is the whole point of free software. To achieve these goals, section 1 states that users of a program must receive both the source code, so that they can make changes to it, and complete documentation, so that they can make informed decisions and give informed consent to modification. To ensure compliance, the section goes on to require that users receive both the source code and documentation from subsequent recipients.

The second goal is to ensure that recipients of a program will have access to all relevant documentation under freedom-including free access to source code. This requires that users receive all relevant documentation that comes with a program and that those users are able to pass it on to subsequent recipients. In the vertical distribution case, this simply means providing users with the source code and with permission to pass verbatim copies along. Thehorizontal distribution case is more complicated, because the ability to pass documentation freely to others is paramount.

The third goal is to protect users from supporting a version of the program that is wrong or potentially harmful. Therefore, users receive source code in order to be able to check that it contains exactly what the author intended. This also allows the user to update the program's authoring tools, such as makefiles or test programs, so that the program can be compiled without errors.

The fourth goal is to permit running the modified version. This is a much simpler requirement than it might at first appear. In order for a modified version to be freely available, users must be able to run it. This requires that users be able to modify source code, so that they can translate it into machine language.

Finally, the fifth goal is to protect the program from unwanted modification. This requires that users be able to check that what was given to them is exactly what the author delivered. It also requires that future recipients be able to verify that what they receive is the same as what they are given. This requires that users have access to the corresponding source code.

The GPL accomplishes these goals by defining specific terms and conditions that must be met in order for a modified version to be freely available. The most important of these terms and conditions are:

- Users must receive the complete source code of the program they receive, including any changes made to the program. This means that any changes to the program must be made available to all subsequent recipients.
- Users must also receive the corresponding documentation, including manuals and any other documentation that comes with the program. This means that any changes to the documentation must also be made available to all subsequent recipients.
- Users must be able to run the program as it was delivered to them.

These terms and conditions ensure that users have access to the source code, documentation, and the ability to run the program, which is necessary for them to modify and distribute it freely.

The GPL also includes provisions for handling cases where the user is not the owner of the computer, and for ensuring that source code is delivered along with any derived works. It also includes provisions for dealing with cases where the program is distributed over the Internet, and for ensuring that users have access to the source code.

By enforcing these terms and conditions, the GPL ensures that the free software movement can continue to grow and thrive, and that users have the freedom to make the software they use as useful as possible.

The GPL allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software. It also ensures that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely.

In summary, the GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a crucial component of the free software movement, and its success is essential for the continued growth and development of free software. It provides a clear and enforceable set of terms and conditions that ensure that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.

The GPL is a powerful tool for ensuring that users have access to the source code, documentation, and the ability to run the software, which is necessary for them to be able to modify and distribute it freely. It also allows users to create and share modified versions of the software, and to receive source code and documentation, which is necessary for them to be able to make informed decisions about the software.
Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language.
(Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on
the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a
storage or distribution medium does not bring the other work under the
scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable
      source code, which must be distributed under the terms of Sections 1
      and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three
      years, to give any third party, for a charge no more than your cost
      of physically performing source distribution, a complete
      machine-readable copy of the corresponding source code, to be
      distributed under the terms of Sections 1 and 2 above on a medium
      customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to
      distribute corresponding source code. (This alternative is allowed
      only for noncommercial distribution and only if you received the
      program in object code or executable form with such an offer, in
      accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program’s name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

### CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None
## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.113 libpng 1.6.37

1.113.1 Available under license :

```
Copyright (c) 1998-2008 Greg Roelofs. All rights reserved.

This software is provided "as is," without warranty of any kind, express or implied. In no event shall the author or contributors be held liable for any damages arising in any way from the use of this software.

The contents of this file are DUAL-LICENSED. You may modify and/or redistribute this software according to the terms of one of the following two licenses (at your option):

LICENSE 1 ("BSD-like with advertising clause"):

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. Redistributions of source code must retain the above copyright notice, disclaimer, and this list of conditions.
2. Redistributions in binary form must reproduce the above copyright notice, disclaimer, and this list of conditions in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   This product includes software developed by Greg Roelofs and contributors for the book, "PNG: The Definitive Guide," published by O'Reilly and Associates.

LICENSE 2 (GNU GPL v2 or later):
```

---

---
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

pnm2png / png2pnm --- conversion from PBM/PGM/PPM-file to PNG-file

copyright (C) 1999-2019 by Willem van Schaik <willem at schaik dot com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the software or the use or other dealings in the software.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE

PNG Reference Library License version 2

* Copyright (c) 1995-2019 The PNG Reference Library Authors.
* Copyright (c) 2018-2019 Cosmin Truta.
* Copyright (c) 1996-1997 Andreas Dilger.
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabuddhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:
There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

   Tom Lane
   Glenn Randers-Pehrson
   Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

   John Bowler
   Kevin Bracey
   Sam Bushell
   Magnus Holmgren
   Greg Roelofs
   Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

   Andreas Dilger
   Dave Martindale
   Guy Eric Schalnat
   Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.

3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>  

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.114 hk2-locator 2.6.1
1.114.1 Available under license:
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the
contents of the Program, including, for purposes of clarity any new file
in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely
in each case in order to link to, bind by name, or subclass the Program
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available
in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making
modifications, including but not limited to software source code,
documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,
Version 2.0, or any later versions of that license, including any
exceptions or additional permissions as identified by the initial
2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS
3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (“notices”) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its
Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General
Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language.
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

   (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on
the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a
storage or distribution medium does not bring the other work under the
scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections 1
and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your cost
of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

 c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program’s name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None
## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.115 kubernetes-client 5.4.1

1.115.1 Available under license:

No license file was found, but licenses were detected in source scan.

```java
/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/lib/FilenameUtils.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/lib/FileSystem.java

No license file was found, but licenses were detected in source scan.

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Copyright Open Source Used In IOx-FogDirector-AC18 1.17.0 1087
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/META-INF/maven/io.fabric8/kubernetes-client/pom.xml
No license file was found, but licenses were detected in source scan.

# # Copyright (C) 2015 Red Hat, Inc.
# # Licensed under the Apache License, Version 2.0 (the "License"); # you may not use this file except in compliance with the License. # You may obtain a copy of the License at
# # http://www.apache.org/licenses/LICENSE-2.0
# # Unless required by applicable law or agreed to in writing, software # distributed under the License is distributed on an "AS IS" BASIS, # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. # See the License for the specific language governing permissions and # limitations under the License.
#

io.fabric8.kubernetes.client.URLFromEnvVarsImpl
io.fabric8.kubernetes.client.URLFromIngressImpl
io.fabric8.kubernetes.client.URLFromNodePortImpl
io.fabric8.kubernetes.client.URLFromClusterIPImpl

Found in path(s):
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/META-INF/services/io.fabric8.kubernetes.client.ServiceToURLProvider
No license file was found, but licenses were detected in source scan.

/** *
* Copyright (C) 2015 Red Hat, Inc.
* *
* Licensed under the Apache License, Version 2.0 (the "License"); *
* you may not use this file except in compliance with the License. *
* You may obtain a copy of the License at *
* *
* http://www.apache.org/licenses/LICENSE-2.0 *
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/informers/impl/DefaultSharedIndexInformer.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/PortForward.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/V1AuthorizationAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Timeable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Recreateable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/VisitFromServerWritable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/V1AutoscalingAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/HttpClientAware.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Parameterizable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/WatchAndWaitable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/V1AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Rollable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/PropagationPolicyConfigurable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/V1CertificatesAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/FlowControlAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Triggerable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Cascading.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/internal/PodMetricOperationsImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/FromServerGettable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes/client/dsl/Replaceable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-jar/io/fabric8/kubernetes клиент/dsl/Replaceable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/CustomResourceList.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/TtyExecInputOutputErroorable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/BatchAPIGroupDSL.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/RawRequestBuilder.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/ListVisitFromServerWritable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/Createable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/batch/v1beta1/CronJobOperationsImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/AutoscalingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/ClusterOperationsImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/informers/ResyncRunnable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/V1beta1DiscoveryAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/URLFromNodePortImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/extended/run/RunConfigUtil.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/DeploymentOperationsImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/utils/KubernetesVersionFactory.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/NamespaceVisitFromServerGetWatchDeleteRecreateWaitApplicableImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/ServiceToURLProvider.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/extended/leaderelection/resourcelock/LeaderElectionRecord.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/internal/RawWatchConnectionManager.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/ExecWatch.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/Scaleable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/utils/BackwardsCompatibilityInterceptor.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/ExecWatch.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/dsl/Scaleable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources.jar/io/fabric8/kubernetes/client/utils/BackwardsCompatibilityInterceptor.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/utils/ImpersonatorInterceptor.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/ReplicaSetRollingUpdater.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/Executable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/V1CertificatesAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/utils/OpenIDConnectionUtils.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/utils/ReflectUtils.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/base/BaseOperation.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/internal/extensions/v1beta1/ReplicaSetOperationsImpl.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/Secretable.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/SchedulingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderElector.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/DiscoveryAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/ScalableResource.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/V2beta2AutoscalingAPIGroupDSL.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/internal/PatchUtils.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/extended/leaderelection/LeaderElectionConfig.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/internal/CustomResourceOperationContext.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/CertificatesAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/V1beta1FlowControlAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/V1beta1SchedulingAPIGroupExtensionAdapter.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/internal/apps/v1/StatefulSetRollingUpdater.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/V1NetworkAPIGroupClient.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/dsl/V1DiscoveryAPIGroupDSL.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
jar/io/fabric8/kubernetes/client/utils/BlockingInputStreamPumper.java
* /opt/cola/permits/1283431052_1646721346.01/0/kubernetes-client-5-4-1-sources-
No license file was found, but licenses were detected in source scan.

* Copyright (C) 2015 Red Hat, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

#
# Copyright (C) 2015 Red Hat, Inc.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
io.fabric8.kubernetes.client.AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1AdmissionRegistrationAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V2beta1AutoscalingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1BatchAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1beta1BatchAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.SchedulingAPIGroupExtensionAdapter
io.fabric8.kubernetes.client.V1SchedulingAPIGroupExtensionAdapter
1.116 giflib 5.2.1

1.116.1 Available under license:

   The GIFLIB distribution is Copyright (c) 1997 Eric S. Raymond

   Permission is hereby granted, free of charge, to any person obtaining a copy
   of this software and associated documentation files (the "Software"), to deal
   in the Software without restriction, including without limitation the rights
   to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
   copies of the Software, and to permit persons to whom the Software is
   furnished to do so, subject to the following conditions:

   The above copyright notice and this permission notice shall be included in
   all copies or substantial portions of the Software.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
   AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
   LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
   OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
   THE SOFTWARE.

1.117 netty-handler-proxy 4.1.49.Final
1.117.1 Available under license:

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497266_1594417706.27/0/netty-handler-proxy-4-1-49-final-sources-1-
jar/META-INF/maven/io.netty/netty-handler-proxy/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497266_1594417706.27/0/netty-handler-proxy-4-1-49-final-sources-1-
jar/io/netty/handler/proxy/ProxyConnectionEvent.java
* /opt/ws_local/PERMITS_SQL/1068497266_1594417706.27/0/netty-handler-proxy-4-1-49-final-sources-1-
jar/io/netty/handler/proxy/HttpProxyHandler.java
* /opt/ws_local/PERMITS_SQL/1068497266_1594417706.27/0/netty-handler-proxy-4-1-49-final-sources-1-
jar/io/netty/handler/proxy/package-info.java
* /opt/ws_local/PERMITS_SQL/1068497266_1594417706.27/0/netty-handler-proxy-4-1-49-final-sources-1-
jar/io/netty/handler/proxy/Socks4ProxyHandler.java
1.118 jackson-dataformat-yaml 2.11.2

1.118.1 Available under license:
# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

## Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (http://fasterxml.com).

## Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.119 spring-jdbc 2.5.6

1.119.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing
the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.120 shiro-spring 1.7.1
1.120.1 Available under license :

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, 
or other liability obligations and/or rights consistent with this 
License. However, in accepting such obligations, You may act only 
on Your own behalf and on Your sole responsibility, not on behalf 
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.121 saaj 1.5.2

1.121.1 Available under license :

==================================================================================================
== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0. ==
== in this case for the Apache Axis2 distribution. ==
==================================================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Portions Copyright 2006 International Business Machines Corp.
Portions Copyright 2005-2007 WSO2, Inc.

This product also includes schemas and specification developed by:
- the W3C consortium (http://www.w3c.org)

This product also includes WS-* schemas developed by International Business Machines Corporation, Microsoft Corporation, BEA Systems, TIBCO Software, SAP AG, Sonic Software, and VeriSign

This product also includes a WSDL developed by salesforce.com
- Copyright 1999-2006 salesforce.com, inc.

Portions of the included xmlbeans library were originally based on the following:

Please read the different LICENSE files present in the lib directory of this distribution.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.122 spring-aspects 5.2.9.RELEASE

1.122.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-
  jar/org/springframework/transaction/aspectj/AbstractTransactionAspect.aj
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-
  jar/org/springframework/beans/factory/aspectj/GenericInterfaceDrivenDependencyInjectionAspect.aj
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-
  jar/org/springframework/cache/aspectj/AnnotationCacheAspect.aj
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1.jar/org/springframework/cache/aspectj/AbstractCacheAspect.aj
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1.jar/org/springframework/beans.factory/aspectj/AbstractDependencyInjectionAspect.aj

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/scheduling/aspectj/AbstractAsyncExecutionAspect.aj
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/scheduling/aspectj/AnnotationAsyncExecutionAspect.aj

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2020 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/transaction/aspectj/AspectJttaTransactionManagementConfiguration.java
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/beans/factory/aspectj/AbstractInterfaceDrivenDependencyInjectionAspect.aj
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/transaction/aspectj/AspectJTransactionManagementConfiguration.java
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/scheduling/aspectj/AspectJAsyncConfiguration.java
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/beans/factory/aspectj/AnnotationBeanConfigurerAspect.aj

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
*   https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/transaction/aspectj/AnnotationTransactionAspect.aj
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/transaction/aspectj/JtaAnnotationTransactionAspect.aj
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2012 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*      https://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/beans/factory/aspectj/ConfigurableObject.java
* /opt/cola/permits/1135968180_1613642617.85/0/spring-aspects-5-2-9-release-sources-1-jar/org/springframework/context/annotation/aspectj/EnableSpringConfigured.java

1.123 shiro-crypto-hash 1.7.1

1.123.1 Available under license :
Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
    stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
you Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.124 vertx-unit 3.9.4

1.124.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.125 activation 1.2.0

1.125.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b)
1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For
purposes of this definition, "control" means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject
to third party intellectual property claims, the Initial Developer
hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or
trademark) Licensable by Initial Developer, to use, reproduce,
modify, display, perform, sublicense and distribute the Original
Software (or portions thereof), with or without Modifications,
and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of
Original Software, to make, have made, use, practice, sell, and
offer for sale, and/or otherwise dispose of the Original Software
(or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on
the date Initial Developer first distributes or otherwise makes the
Original Software available to a third party under the terms of this
License.

(d) Notwithstanding Section 2.1(b) above, no patent license is
granted: (1) for code that You delete from the Original Software, or
(2) for infringements caused by: (i) the modification of the
Original Software, or (ii) the combination of the Original Software
with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject
to third party intellectual property claims, each Contributor hereby
grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or
trademark) Licensable by Contributor to use, reproduce, modify,
display, perform, sublicense and distribute the Modifications
created by such Contributor (or portions thereof), either on an
unmodified basis, with other Modifications, as Covered Software
and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not
remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients’ rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to
cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH
PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such
responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

-----------------------------------------------

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously
and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in
full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED...
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#

Certain source files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

You should also note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items
included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.126 metrics-json 4.1.9
1.126.1 Available under license:
   Apache-2.0

1.127 pcre 8.39-9
1.127.1 Available under license:
   PCRE LICENCE
   1

PCRE is a library of functions to support regular expressions whose syntax and semantics are as close as possible to those of the Perl 5 language.

Release 8 of PCRE is distributed under the terms of the "BSD" licence, as specified below. The documentation for PCRE, supplied in the "doc" directory, is distributed under the same terms as the software itself. The data in the testdata directory is not copyrighted and is in the public domain.
The basic library functions are written in C and are freestanding. Also included in the distribution is a set of C++ wrapper functions, and a just-in-time compiler that can be used to optimize pattern matching. These are both optional features that can be omitted when the library is built.

THE BASIC LIBRARY FUNCTIONS
----------------------------

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,

Copyright (c) 1997-2016 University of Cambridge
All rights reserved.

PCRE JUST-IN-TIME COMPILATION SUPPORT
-------------------------------------

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2010-2016 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER
--------------------------------

Written by: Zoltan Herczeg
Email local part: hzmester
Email domain: freemail.hu

Copyright(c) 2009-2016 Zoltan Herczeg
All rights reserved.

THE C++ WRAPPER FUNCTIONS
----------------------------

Contributed by: Google Inc.
THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the copyright notice, this list of conditions and the following disclaimer in the
Open Source Used In IOx-FogDirector-AC18 1.17.0

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.128 ejb-api 3.0-alpha-1

1.128.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or
portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to
make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish),
that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new
free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to
surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all
the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show
them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you
legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no
warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to
know that what they have is not the original, so that any problems introduced by others will not reflect on the
original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that
redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To
prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

 TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying
it may be distributed under the terms of this General Public License. The "Program", below, refers to any such
program or work, and a "work based on the Program" means either the Program or any derivative work under
copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications
and/or translated into another language. (Hereinafter, translation is included without limitation in the term
"modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its
scope. The act of running the Program is not restricted, and the output from the Program is covered only if its
contents constitute a work based on the Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium,
provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and
give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty
protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted
interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special
exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.129 perl 5.26.1 6ubuntu0.5

1.129.1 Available under license:

--------------------------------------------------------------------------------------------------------------------------

This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2010 Julian R Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jsward@bzip.org
bzip2/libbzip2 version 1.0.6 of 6 September 2010

---
abstract: 'Build and install Perl modules'
author:
  - 'Ken Williams <kwilliams@cpan.org>'
  - "Development questions, bug reports, and patches should be sent to the
Module-Build mailing list at <module-build@perl.org>.
"
built_requires:
  File::Temp: 0.15
  Test::Harness: 3.16
  Test::More: 0.49
generated_by: 'Module::Build version 0.3608'
license: gpl
meta-spec:
  url: http://module-build.sourceforge.net/META-spec-v1.4.html
  version: 1.4
name: Module-Build
resources:
  MailingList: mailto:module-build@perl.org
  license: http://dev.perl.org/licenses/
  repository: http://github.com/dagolden/module-build/
  version: 3

The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute
the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.
b) use the modified Package only within your corporation or organization.

c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

b) accompany the distribution with the machine-readable source of the Package with your modifications.

c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.
7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End

GNU GENERAL PUBLIC LICENSE
Version 1, February 1989

Copyright (C) 1989 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The license agreements of most software companies try to keep users at the mercy of those companies. By contrast, our General Public License is intended to guarantee your freedom to share and change free software--to make sure that the software is free for all its users. The General Public License applies to the Free Software Foundation's software and to any other program whose authors commit to using it. You can use it for your programs, too.

When we speak of free software, we are referring to freedom, not price. Specifically, the General Public License is designed to make sure that you have the freedom to give away or sell copies of free software, that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free
programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of a such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any work containing the Program or a portion of it, either verbatim or with modifications. Each licensee is addressed as "you".

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this General Public License and to the absence of any warranty; and give any other recipients of the Program a copy of this General Public License along with the Program. You may charge a fee for the physical act of transferring a copy.

2. You may modify your copy or copies of the Program or any portion of it, and copy and distribute such modifications under the terms of Paragraph
I above, provided that you also do the following:

a) cause the modified files to carry prominent notices stating that
you changed the files and the date of any change; and

b) cause the whole of any work that you distribute or publish, that
in whole or in part contains the Program or any part thereof, either
with or without modifications, to be licensed at no charge to all
third parties under the terms of this General Public License (except
that you may choose to grant warranty protection to some or all
third parties, at your option).

c) If the modified program normally reads commands interactively when
run, you must cause it, when started running for such interactive use
in the simplest and most usual way, to print or display an
announcement including an appropriate copyright notice and a notice
that there is no warranty (or else, saying that you provide a
warranty) and that users may redistribute the program under these
conditions, and telling the user how to view a copy of this General
Public License.

d) You may charge a fee for the physical act of transferring a
copy, and you may at your option offer warranty protection in
exchange for a fee.

Mere aggregation of another independent work with the Program (or its
derivative) on a volume of a storage or distribution medium does not bring
the other work under the scope of these terms.

3. You may copy and distribute the Program (or a portion or derivative of
it, under Paragraph 2) in object code or executable form under the terms of
Paragraphs 1 and 2 above provided that you also do one of the following:

a) accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of
Paragraphs 1 and 2 above; or,

b) accompany it with a written offer, valid for at least three
years, to give any third party free (except for a nominal charge
for the cost of distribution) a complete machine-readable copy of the
corresponding source code, to be distributed under the terms of
Paragraphs 1 and 2 above; or,

c) accompany it with the information you received as to where the
corresponding source code may be obtained. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form alone.)
Source code for a work means the preferred form of the work for making modifications to it. For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs, or for standard header files or definitions files that accompany that operating system.

4. You may not copy, modify, sublicense, distribute or transfer the Program except as expressly provided under this General Public License. Any attempt otherwise to copy, modify, sublicense, distribute or transfer the Program is void, and will automatically terminate your rights to use the Program under this License. However, parties who have received copies, or rights to use copies, from you under this General Public License will not have their licenses terminated so long as such parties remain in full compliance.

5. By copying, distributing or modifying the Program (or any work based on the Program) you indicate your acceptance of this license to do so, and all its terms and conditions.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

7. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of the license which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the license, you may choose any version ever published by the Free Software Foundation.

8. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

9. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

10. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to humanity, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston MA  02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19xx name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the
commands you use may be called something other than `show w' and `show
c'; they could even be mouse-clicks or menu items--whatever suits your
program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (a program to direct compilers to make passes
at assemblers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

That's all there is to it!
This software is copyright (c) 2017 by Ken Williams.

This is free software; you can redistribute it and/or modify it under
the same terms as the Perl 5 programming language system itself.

Terms of the Perl programming language system itself

a) the GNU General Public License as published by the Free
   Software Foundation; either version 1, or (at your option) any
   later version, or
b) the "Artistic License"

--- The GNU General Public License, Version 1, February 1989 ---
This software is Copyright (c) 2017 by Ken Williams.

This is free software, licensed under:

The GNU General Public License, Version 1, February 1989

GNU GENERAL PUBLIC LICENSE
Version 1, February 1989

Copyright (C) 1989 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The license agreements of most software companies try to keep users at the mercy of those companies. By contrast, our General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. The General Public License applies to the Free Software Foundation's software and to any other program whose authors commit to using it. You can use it for your programs, too.

When we speak of free software, we are referring to freedom, not price. Specifically, the General Public License is designed to make sure that you have the freedom to give away or sell copies of free software, that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of a such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any work containing the Program or a portion of it, either verbatim or with modifications. Each licensee is addressed as "you".

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this General Public License and to the absence of any warranty; and give any other recipients of the Program a copy of this General Public License along with the Program. You may charge a fee for the physical act of transferring a copy.

2. You may modify your copy or copies of the Program or any portion of it, and copy and distribute such modifications under the terms of Paragraph 1 above, provided that you also do the following:

   a) cause the modified files to carry prominent notices stating that you changed the files and the date of any change; and

   b) cause the whole of any work that you distribute or publish, that in whole or in part contains the Program or any part thereof, either with or without modifications, to be licensed at no charge to all third parties under the terms of this General Public License (except that you may choose to grant warranty protection to some or all third parties, at your option).

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the simplest and most usual way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these
conditions, and telling the user how to view a copy of this General Public License.

d) You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another independent work with the Program (or its derivative) on a volume of a storage or distribution medium does not bring the other work under the scope of these terms.

3. You may copy and distribute the Program (or a portion or derivative of it, under Paragraph 2) in object code or executable form under the terms of Paragraphs 1 and 2 above provided that you also do one of the following:

a) accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Paragraphs 1 and 2 above; or,

b) accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal charge for the cost of distribution) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Paragraphs 1 and 2 above; or,

c) accompany it with the information you received as to where the corresponding source code may be obtained. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form alone.)

Source code for a work means the preferred form of the work for making modifications to it. For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs, or for standard header files or definitions files that accompany that operating system.

4. You may not copy, modify, sublicense, distribute or transfer the Program except as expressly provided under this General Public License. Any attempt otherwise to copy, modify, sublicense, distribute or transfer the Program is void, and will automatically terminate your rights to use the Program under this License. However, parties who have received copies, or rights to use copies, from you under this General Public License will not have their licenses terminated so long as such parties remain in full compliance.

5. By copying, distributing or modifying the Program (or any work based
(on the Program) you indicate your acceptance of this license to do so, and all its terms and conditions.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

7. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of the license which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the license, you may choose any version ever published by the Free Software Foundation.

8. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

9. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

10. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to humanity, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

   Gnomovision version 69, Copyright (C) 19xx name of author
   Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
   This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the
commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (a program to direct compilers to make passes at assemblers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

That's all there is to it!

--- The Artistic License 1.0 ---

This software is Copyright (c) 2017 by Ken Williams.

This is free software, licensed under:

The Artistic License 1.0

The Artistic License

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

- "Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.
- "Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder.
- "Copyright Holder" is whoever is named in the copyright or copyrights for the package.
- "You" is you, if you're thinking about copying or distributing this Package.
- "Reasonable copying fee" is whatever you can justify on the basis of media
- “Freely Available” means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as ftp.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

   b) use the modified Package only within your corporation or organization.

   c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

   a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

   b) accompany the distribution with the machine-readable source of the Package with your modifications.

   c) accompany any non-standard executables with their corresponding Standard Version executables, giving the non-standard executables non-standard names, and clearly documenting the differences in manual pages (or
equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this Package.

7. C or perl subroutines supplied by you and linked into this Package shall not be considered part of this Package.

8. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

9. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End
#!perl

=head1 NAME

copyright.t

=head1 DESCRIPTION

Tests that the latest copyright years in the top-level README file and the C<perl -v> output match each other.

If the test fails, update at least one of README and perl.c so that they match reality.

Optionally you can pass the C<--now> option to check they are at the current year. This isn't checked by default, so that it doesn't fail for people working on older releases. It should be run before making a new release.

=cut
use strict;
use Config;
BEGIN { require './test.pl' }

if ( $Config{usecrosscompile} ) {
    skip_all( "Not all files are available during cross-compilation" );
}

my ($opt) = @ARGV;

my $readme_year = readme_year();
my $v_year = v_year();

# Check that both copyright dates are up-to-date, but only if requested, so
# that tests still pass for people intentionally working on older versions:
if ($opt eq '--now') {
    my $current_year = (gmtime)[5] + 1900;
    is $v_year, $current_year, 'perl -v copyright includes current year';
    is $readme_year, $current_year, 'README copyright includes current year';
}

# Otherwise simply check that the two copyright dates match each other:
else {
    is $readme_year, $v_year, 'README and perl -v copyright dates match';
}

done_testing;

sub readme_year
# returns the latest copyright year from the top-level README file
{
    open my $readme, '<', '../README' or die "Opening README failed: $!";

    # The copyright message is the first paragraph:
    local $/ = ";
    my $copyright_msg = <$readme>;

    my ($year) = $copyright_msg =~ /.*\b\d{4,}\b/; # or die "Year not found in README copyright message \'$copyright_msg\'";

    $year;
}
sub v_year
# returns the latest copyright year shown in perl -v
{
    my $output = runperl switches => ['-v'];
    my ($year) = $output =~ /copyright 1987.*(\d{4,})/i
        or die "Copyright statement not found in perl -v output '$output'";

    $year;
}

## 1.130 libffi 3.2.1 8
### 1.130.1 Available under license:

libffi - Copyright (c) 1996-2014 Anthony Green, Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
``Software``), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This package was debianized by Matthias Klose <doko@debian.org>
on Mon, 10 Mar 2008 05:46:40 +0000.

It was downloaded from ftp://sourceware.org/pub/libffi/

Upstream Authors:

Anthony Green <green@redhat.com>
GCC developers

See the README and below in the list of copyright holders for a more
complete list.
Copyright:
Copyright (c) 1996-2011 Red Hat, Inc.
Copyright (C) 1996-2011 Anthony Green
Copyright (C) 1996-2010 Free Software Foundation, Inc
Copyright (c) 2010, 2011, Plausible Labs Cooperative, Inc.
Copyright (c) 2010 CodeSourcery
Copyright (c) 1998 Andreas Schwab
Copyright (c) 2000 Hewlett Packard Company
Copyright (c) 2009 Bradley Smith
Copyright (c) 2008 David Daney
Copyright (c) 2004 Simon Posnjak
Copyright (c) 2005 Axis Communications AB
Copyright (c) 1998 Cygnus Solutions
Copyright (c) 2004 Renesas Technology
Copyright (c) 2002, 2007 Bo Thorsen <bo@suse.de>
Copyright (c) 2002 Ranjit Mathew
Copyright (c) 2002 Roger Sayle
Copyright (c) 2000, 2007 Software AG
Copyright (c) 2003 Jakub Jelinek
Copyright (c) 2000, 2001 John Hornkvist
Copyright (c) 1998 Geoffrey Keating
Copyright (c) 2008 Bjrn Knig

License:

libffi - Copyright (c) 1996-2010 Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.131 libsigar 1.6.5

1.131.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined bySections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized bythe copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all otherentities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct orindirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership offifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software sourcecode, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form,
including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.132 xstream 1.4.17

1.132.1 Available under license:

No license file was found, but licenses were detected in source scan.

Copyright (C) 2006 Joe Walnes.
XStream committers.
All rights reserved.

The software in this package is published under the terms of the BSD
style license a copy of which has been included with this distribution in
the LICENSE.txt file.

Created on 29. July 2006 by Mauro Talevi

Found in path(s):
* /opt/cola/permits/1166458259_1621426611.06/0/xstream-1-4-17-jar/META-INF/maven/com.thoughtworks.xstream/xstream/pom.xml
No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Archiver-Version: Plexus Archiver
1.133 antlr 2.7.7

1.133.1 Available under license:

SOFTWARE RIGHTS

ANTLR 1989-2006 Developed by Terence Parr
Partially supported by University of San Francisco & jGuru.com

We reserve no legal rights to the ANTLR--it is fully in the public domain. An individual or company may do whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs...
(commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

The primary ANTLR guy:

Terence Parr
partr@cs.usfca.edu
partr@antlr.org

1.134 jsr305 3.0.2

1.134.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
  * Copyright (c) 2005 Brian Goetz
  * Released under the Creative Commons Attribution License
  * (http://creativecommons.org/licenses/by/2.5)
  * Official home: http://www.jcip.net
  */

Found in path(s):
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/GuardedBy.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/ThreadSafe.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/Immutable.java
* /opt/cola/permits/1135861076_1613617584.8/0/jsr305-3-0-2-sources-4-jar/javax/annotation/concurrent/NotThreadSafe.java

1.135 swagger-annotations 1.5.20

1.136 jaxb-api 2.3.1
1.136.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty
percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this
License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as
"Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-
law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

---------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this
License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.
One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.137 xml-commons-resolver 1.2
1.137.1 Available under license:

LEGAL NOTICE

This notice is issued to fulfil the requirements of the Mozilla Public License version 1.0 ("MPL 1.0") sections 3.4(a) and 3.6. MPL 1.0 can be found at http://www.mozilla.org/MPL/MPL-1.0.html.

Section 3.4(a) of MPL 1.0 states that any third party intellectual property rights in particular functionality or code must be notified in a text file named LEGAL that is issued with the source code. Saxon includes a number of such third party components, and the relevant claims are included in notices included in the same directory as this notice. Although MPL 1.0 requires this notice to be included only with source code, some of the third parties may also require notices to be included with executable code. Therefore, Saxon executable code must not be distributed separately from this notice and all the accompanying third party notices. The term "Distribution" here includes making the code available for download, and its inclusion in download repositories such as Maven.

Section 3.6 of MPL 1.0 states:

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2.

Section 3.2 requires the Source Code of Covered Code to be made available via an accepted Electronic Distribution Mechanism.

The Source Code version of the Covered Code (that is, the source code of Saxon-B) is available under the terms of the Mozilla Public License version 1.0, and may be obtained from the Subversion repository for the Saxon project on SourceForge, at https://sourceforge.net/svn/?group_id=29872. The precise version of the Subversion source for a particular Saxon maintenance release can be determined by referring to the release notes for the particular release in the SourceForge download area.

Note that MPL 1.0 requires that any modifications to this source code must be made available under the terms of the MPL "to anyone to whom you made an executable version available". As a courtesy, it is also requested
that you make such modifications available to Saxonica Limited.
The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license
document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share
and change it. By contrast, the GNU General Public License is intended to
guarantee your freedom to share and change free software--to make sure the
software is free for all its users. This General Public License applies to
most of the Free Software Foundation's software and to any other program whose
authors commit to using it. (Some other Free Software Foundation software is
covered by the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not price. Our
General Public Licenses are designed to make sure that you have the freedom to
distribute copies of free software (and charge for this service if you wish),
that you receive source code or can get it if you want it, that you can change
the software or use pieces of it in new free programs; and that you know you
can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny
you these rights or to ask you to surrender the rights. These restrictions
translate to certain responsibilities for you if you distribute copies of the
software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for
a fee, you must give the recipients all the rights that you have. You must
make sure that they, too, receive or can get the source code. And you must
show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2)
offer you this license which gives you legal permission to copy, distribute
and/or modify the software.

Also, for each author's protection and ours, we want to make certain that
everyone understands that there is no warranty for this free software. If the
software is modified by someone else and passed on, we want its recipients to
know that what they have is not the original, so that any problems introduced
by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be
Open Source Used In IOx-FogDirector-AC18 1.17.0

1209

licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only
for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all.
For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of
software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR
THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE
PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND
PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE,
YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL
ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE
THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR
INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA
BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER
OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible
use to the public, the best way to achieve this is to make it free software
which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach
them to the start of each source file to most effectively convey the exclusion
of warranty; and each file should have at least the "copyright" line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by the Free
Software Foundation; either version 2 of the License, or (at your option)
any later version.

This program is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for
more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL, but only where Sun has expressly included in the particular source file’s header the words "Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making
a combined work based on this library. Thus, the terms and conditions of
the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent modules,
and to copy and distribute the resulting executable under terms of your
choice, provided that you also meet, for each linked independent module,
the terms and conditions of the license of that module. An independent
module is a module which is not derived from or based on this library. If
you modify this library, you may extend this exception to your version of
the library, but you are not obligated to do so. If you do not wish to do
so, delete this exception statement from your version.

Apache XML Commons Resolver
Copyright 2006 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation http://www.apache.org/

Portions of this code are derived from classes placed in the
public domain by Arbortext on 10 Apr 2000. See:
http://www.arbortext.com/customer_support/updates_and_technical_notes/catalogs/docs/README.htm
The contents of these file are subject to the Mozilla Public License Version 1.0 (the "License);
you may not use these files except in compliance with the License. You may obtain a copy of the
License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS" basis,
WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the
specific language governing rights and limitations under the License.

The Original Code is all Saxon modules labelled with a notice referring to this license.

The Initial Developer of the Original Code is Michael Kay, except where otherwise specified in an individual
module.

Portions created by other named contributors are copyright as identified in the relevant module. All Rights
Reserved.

Contributor(s) are listed in the documentation: see notices/contributors.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.138 jta 1.3
1.138.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1. You, (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor
3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.
4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as a Participant,) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day
notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.
10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and
that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your
work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from
the Program, and can be reasonably considered independent and separate works in themselves, then this License,
and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute
the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on
the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and
every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather,
the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on
the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under
      the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more
      than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding
      source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software
      interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This
      alternative is allowed only for noncommercial distribution and only if you received the program in object code or
      executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable
work, complete source code means all the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is normally distributed (in either source or
binary form) with the major components (compiler, kernel, and so on) of the operating system on which the
executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering
equivalent access to copy the source code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License.
Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically
terminate your rights under this License. However, parties who have received copies, or rights, from you under this
License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your
version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.139 libnss 3.35-2ubuntu2.13

1.139.1 Available under license:

This package was debianized by Mike Hommey <glandium@debian.org> on Sun, 25 Mar 2007 19:36:42 +0200.

It was downloaded from http://ftp.mozilla.org/pub/mozilla.org/security/nss/

Upstream Author: The Mozilla Project.

The NSS library is licensed under the terms of the Mozilla Public License version 2.0, which terms can be found further below.

The original code is copyright (c) 1994-2000 Netscape Communications Corporation.

Some external libraries are also provided in the source tree with the following licensing terms:

=== zlib

(C) 1995-2004 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu
If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes. Please read the FAQ for more information on the distribution of modified source versions.

=== dbm

The nss/lib/dbm directory, with few exceptions, is licensed under the following terms:

Copyright (c) 1991, 1993, 1994
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=== sqlite
The nss/lib/sqlite/sqlite3.[ch] files contain a copy of sqlite with the following licensing terms:

The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

    May you do good and not evil.
    May you find forgiveness for yourself and forgive others.
    May you share freely, never taking more than you give.

=== mkdepend

The nss/coreconf/mkdepend directory contains a copy of mkdepend with the following licensing terms:

cppsetup.c, def.h, include.c, main.c, mkdepend.man, parse.c, pr.c:
Copyright (c) 1993, 1994, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

ifparser.[ch]:

Copyright 1992 Network Computing Devices, Inc.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Network Computing Devices may not be used in advertising or publicity pertaining to distribution of the software.
without specific, written prior permission. Network Computing Devices makes
no representations about the suitability of this software for any purpose.
It is provided “as is” without express or implied warranty.

NETWORK COMPUTING DEVICES DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS,
IN NO EVENT SHALL NETWORK COMPUTING DEVICES BE LIABLE FOR ANY SPECIAL,
INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM
LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

imakemdep.h:

Copyright (c) 1993, 1994 X Consortium

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be
used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization from the X Consortium.

=== MPL

Note on GPL Compatibility

The MPL 2, section 3.3, permits you to combine NSS with code under the GNU
General Public License (GPL) version 2, or any later version of that
license, to make a Larger Work, and distribute the result under the GPL.
The only condition is that you must also make NSS, and any changes you
have made to it, available to recipients under the terms of the MPL 2 also.
Anyone who receives the combined code from you does not have to continue to dual licence in this way, and may, if they wish, distribute under the terms of either of the two licences - either the MPL alone or the GPL alone. However, we discourage people from distributing copies of NSS under the GPL alone, because it means that any improvements they make cannot be reincorporated into the main version of NSS. There is never a need to do this for license compatibility reasons.

Note on LGPL Compatibility
--------------------------

The above also applies to combining MPLed code in a single library with code under the GNU Lesser General Public License (LGPL) version 2.1, or any later version of that license. If the LGPLed code and the MPLed code are not in the same library, then the copyleft coverage of the two licences does not overlap, so no issues arise.

Mozilla Public License Version 2.0
==================================

1. Definitions
-------------

1.1. "Contributor"  
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"  
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"  
means Covered Software of a particular Contributor.

1.4. "Covered Software"  
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"  
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:
(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

-----------------------------------
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                          *
*  -------------------------                                          *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a
* particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. * Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

************************************************************************

************************************************************************

7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

----------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

---------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.
You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.140 management-api 1.1-rev-1

1.140.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally
1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions
thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients, rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms
which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software, (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the
jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the
software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of
any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time
to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.141 metrics-spring 3.1.3
1.141.1 Available under license:
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by the Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 1254
institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2012 Ryan Tenney and Martello Technologies Corp.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.142 fastjson 1.2.26
1.142.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
/
* Copyright 1999-2011 Alibaba Group.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

1.143 commons-text 1.9

1.143.1 Available under license:

Apache Commons Text
Copyright 2014-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.144 profiler 1.1.1
1.144.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * jQuery Templates Plugin 1.0.0pre
 * http://github.com/jquery/jquery-tmpl
 * Requires jQuery 1.4.2
 *
 * Copyright Software Freedom Conservancy, Inc.
 * Dual licensed under the MIT or GPL Version 2 licenses.
 * http://jquery.org/license
 */

Found in path(s):
* /opt/cola/permits/1257333100_1642801922.15/0/gae-mini-profiler-1-1-1-sources.jar/ca/jimr/gae/profiler/resources/jquery.tmpl.min.js

No license file was found, but licenses were detected in source scan.
/*!
 * jQuery JavaScript Library v1.6.2
 * http://jquery.com/
 *
 * Copyright 2011, John Resig
 * Dual licensed under the MIT or GPL Version 2 licenses.
 * http://jquery.org/license
 *
 * Includes Sizzle.js
 * http://sizzlejs.com/
 * Copyright 2011, The Dojo Foundation
 * Released under the MIT, BSD, and GPL Licenses.
 *
 * Date: Thu Jun 30 14:16:56 2011 -0400
 */

Found in path(s):
* /opt/cola/permits/1257333100_1642801922.15/0/gae-mini-profiler-1-1-1-sources-
  jar/ca/jimr/gae/profiler/resources/jquery-1.6.2.min.js
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2011 by Jim Riecken
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 * THE SOFTWARE.
 */

Found in path(s):
* /opt/cola/permits/1257333100_1642801922.15/0/gae-mini-profiler-1-1-1-sources-
  jar/com/google/appengine/tools/appstats/MiniProfilerAppstats.java
* /opt/cola/permits/1257333100_1642801922.15/0/gae-mini-profiler-1-1-1-sources-
  jar/ca/jimr/gae/profiler/MiniProfiler.java
1.145 libibverbs1 17.1-1ubuntu0.2

1.145.1 Available under license:

Copyright (c) 2013, Intel Corporation. All rights reserved.
Copyright (c) 2007, QLogic Corp. All rights reserved.
Copyright (c) 2005, PathScale, Inc. All rights reserved.

This software is available to you under a choice of one of two licenses. You may choose to be licensed under the terms of the GNU General Public License (GPL) Version 2, available from the file COPYING in the main directory of this source tree, or the OpenIB.org BSD license below:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Patent licenses, if any, provided herein do not apply to combinations of this program with other software, or any other product whatsoever.

OpenIB.org BSD license (MIT variant)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: rdma-core
Upstream-Contact: Doug Ledford <dledford@redhat.com>, Leon Romanovsky <Leon@kernel.org>
Source: https://github.com/linux-rdma/rdma-core

Files: *
Copyright: disclaimed
License: BSD-MIT or GPL-2

Files: debian/*
Copyright: 2008, Genome Research Ltd
    2014, Ana Beatriz Guerrero Lopez <ana@debian.org>
    2015-2016, Jason Gunthorpe <jgunthorpe@obsidianresearch.com>
    2016-2017, Benjamin Drung <benjamin.drung@profitbricks.com>
    2016-2017, Talat Batheesh <talatb@mellanox.com>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
On Debian systems, the full text of the GNU General Public License version 2 can be found in the file `/usr/share/common-licenses/GPL-2'.

Files: CMakeLists.txt
License: BSD-MIT or GPL-2

Files: buildlib/*
    2016-2017 Mellanox Technologies, Inc
License: BSD-MIT or GPL-2

Files: buildlib/fixup-include/rdma-vmw_pvrdma-abih
Copyright: 2012-2016 VMware, Inc.
License: BSD-2-clause or GPL-2

Files: buildlib/fixup-include/stdatomic.h
Copyright: 2011 Ed Schouten <ed@FreeBSD.org>
    David Chisnall <theraven@FreeBSD.org>
License: BSD-2-clause

Files: ccan/*
Copyright: unspecified
License: CC0

Files: ccan/list.*
Copyright: unspecified
License: MIT

Files: ibacm/*
    2013, Mellanox Technologies LTD.
License: BSD-MIT

Files: ibacm/man/*
    ibacm/ibacm.init.in
Copyright: disclaimed
License: BSD-2-clause

Files: ibacm/CMakeLists.txt
    ibacm/ibacm_hosts.data
Copyright: disclaimed
License: BSD-MIT or GPL-2

Files: iwpmd/*
Copyright: 2013-2016, Intel Corporation.
License: BSD-MIT or GPL-2
Files: libibumad/*
Copyright: 2004-2017, Mellanox Technologies Ltd.
   2004, Topspin Corporation.
   2013 Lawrence Livermore National Security
   2013, Oracle and/or its affiliates.
License: BSD-MIT or GPL-2

Files: libibumad/man/*
Copyright: disclaimed
License: BSD-2-clause

Files: libibverbs/*
   2005, PathScale, Inc.
   2005, Mellanox Technologies Ltd.
   2005, Voltaire, Inc.
   2008, Lawrence Livermore National Laboratory.
License: BSD-MIT or GPL-2

Files: libibverbs/man/*
   libibverbs/neigh.h
   libibverbs/nl1_compat.h
   libibverbs/neigh.c
Copyright: disclaimed
License: BSD-2-clause

Files: librdmacm/*
   2005, Ammasso, Inc.
   2005, Voltaire Inc.
   2014-2015, Mellanox Technologies LTD.
License: BSD-MIT or GPL-2

Files: librdmacm/examples/cmtime.c
   librdmacm/examples/rcopy.c
   librdmacm/examples/rdma_client.c
   librdmacm/examples/rdma_server.c
   librdmacm/examples/rdma_xclient.c
   librdmacm/examples/rdma_xserver.c
   librdmacm/examples/riostream.c
   librdmacm/examples/rstream.c
   librdmacm/examples/udpong.c
2014-2015, Mellanox Technologies LTD.
License: BSD-MIT

Files: librdmacm/docs/rsocket
Copyright: disclaimed
License: BSD-2-clause

Files: librdmacm/man/*
Copyright: disclaimed
License: BSD-2-clause

Files: providers/bnxt_re/*
Copyright: 2015-2017, Broadcom Limited and/or its subsidiaries
License: BSD-2-clause or GPL-2

Files: providers/cxgb3/*
   providers/cxgb4/*
Copyright: 2003-2016, Chelsio Communications, Inc.
License: BSD-MIT or GPL-2

Files: providers/hfi1verbs/*
Copyright: 2005 PathScale, Inc.
   2006-2009 QLogic Corporation
   2015 Intel Corporation
License: BSD-3-clause or GPL-2

Files: providers/hns/*
Copyright: 2016, Hisilicon Limited.
License: BSD-MIT or GPL-2

Files: providers/i40iw/*
Copyright: 2015-2016, Intel Corporation.
License: BSD-MIT or GPL-2

Files: providers/ipathverbs/*
Copyright: 2006-2010, QLogic Corp.
   2005, PathScale, Inc.
   2013, Intel Corporation
License: BSD-MIT or GPL-2

Files: providers/mlx4/*
Copyright: 2004-2005, Topspin Communications.
   2005-2007, Cisco, Inc.
   2005-2017, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: providers/mlx5/*
Copyright: 2010-2017, Mellanox Technologies, Inc.
License: BSD-MIT or GPL-2

Files: providers/mlx5/man/*.
Copyright: disclaimed
License: BSD-MIT

Files: providers/mthca/*
Copyright: 2004-2005, Topspin Communications.
2005, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: providers/nes/*
Copyright: 2006-2010, Intel Corporation.
License: BSD-MIT or GPL-2

Files: providers/ocrdma/*
Copyright: 2008-2013, Emulex.
License: BSD-2-clause or GPL-2

Files: providers/qedr/*
Copyright: 2015-2016, QLogic Corporation.
License: BSD-MIT or GPL-2

Files: providers/rxe/*
Copyright: 2009-2011, System Fabric Works, Inc.
2009-2011, Mellanox Technologies Ltd.
2005, PathScale, Inc.
License: BSD-MIT or GPL-2

Files: providers/vmw_pvrdma/*
Copyright: 2012-2016 VMWare, Inc.
License: BSD-2-clause or GPL-2

Files: rdma-ndd/*
Copyright: 2004-2016 Intel Corporation.
License: BSD-MIT or GPL-2

Files: redhat/*
Copyright: 1996-2013, Red Hat, Inc.
License: GPL-2

Files: srp_daemon/*
Copyright: 2005, Topspin Communications.
2006, Cisco Systems, Inc.
2006, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: srp_daemon/srp_daemon.1.in
Copyright: 2006 Mellanox Technologies.
License: CPL-1.0 or BSD-2-clause or GPL-2

Files: srp_daemon/srpd.in
srp_daemon/ibsrpd.in
Copyright: disclaimed
License: BSD-2-clause

Files: util/udma_barrier.h
Copyright: 2005 Topspin Communications.
License: BSD-MIT or GPL-2

License: BSD-MIT
OpenIB.org BSD license (MIT variant)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

License: BSD-2-clause
OpenIB.org BSD license (FreeBSD Variant)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
  distribution.
* Neither the name of Intel Corporation nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: GPL-2
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; version 2 of the License.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text of the GNU General Public License version 2 can be found in the file `/usr/share/common-licenses/GPL-2'.

License: CC0
The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

   the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work; moral rights retained by the original author(s) and/or performer(s); publicity and privacy rights pertaining to a person's image or likeness depicted in a Work; rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below; rights protecting the extraction, dissemination, use and reuse of data in a Work; database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of
databases, and under any national implementation thereof, including any amended
or successor version of such directive); and other similar, equivalent or
corresponding rights throughout the world based on applicable law or treaty,
and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of,
applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and
unconditionally waives, abandons, and surrenders all of Affirmer's Copyright
and Related Rights and associated claims and causes of action, whether now
known or unknown (including existing as well as future claims and causes of
action), in the Work (i) in all territories worldwide, (ii) for the maximum
duration provided by applicable law or treaty (including future time
extensions), (iii) in any current or future medium and for any number of
copies, and (iv) for any purpose whatsoever, including without limitation
commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes
the Waiver for the benefit of each member of the public at large and to the
detriment of Affirmer's heirs and successors, fully intending that such Waiver
shall not be subject to revocation, rescission, cancellation, termination, or
any other legal or equitable action to disrupt the quiet enjoyment of the Work
by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be
judged legally invalid or ineffective under applicable law, then the Waiver
shall be preserved to the maximum extent permitted taking into account
Affirmer's express Statement of Purpose. In addition, to the extent the Waiver
is so judged Affirmer hereby grants to each affected person a royalty-free, non
transferable, non sublicensable, non exclusive, irrevocable and unconditional
license to exercise Affirmer's Copyright and Related Rights in the Work (i) in
all territories worldwide, (ii) for the maximum duration provided by applicable
law or treaty (including future time extensions), (iii) in any current or
future medium and for any number of copies, and (iv) for any purpose
whatsoever, including without limitation commercial, advertising or promotional
purposes (the "License"). The License shall be deemed effective as of the date
CC0 was applied by Affirmer to the Work. Should any part of the License for any
reason be judged legally invalid or ineffective under applicable law, such
partial invalidity or ineffectiveness shall not invalidate the remainder of the
License, and in such case Affirmer hereby affirms that he or she will not (i)
exercise any of his or her remaining Copyright and Related Rights in the Work
or (ii) assert any associated claims and causes of action with respect to the
Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

No trademark or patent rights held by Affirmer are waived, abandoned,
surrendered, licensed or otherwise affected by this document. Affirmer offers
the Work as-is and makes no representations or warranties of any kind
concerning the Work, express, implied, statutory or otherwise, including
without limitation warranties of title, merchantability, fitness for a
particular purpose, non infringement, or the absence of latent or other
defects, accuracy, or the present or absence of errors, whether or not
discoverable, all to the greatest extent permissible under applicable law.
Affirmer disclaims responsibility for clearing rights of other persons that may
apply to the Work or any use thereof, including without limitation any person's
Copyright and Related Rights in the Work. Further, Affirmer disclaims
responsibility for obtaining any necessary consents, permissions or other
rights required for any use of the Work. Affirmer understands and acknowledges
that Creative Commons is not a party to this document and has no duty or
obligation with respect to this CC0 or use of the Work.

License: MIT
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

License: CPL-1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC
LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and
documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;
where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are not
derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement,
including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to
reproduce, prepare derivative works of, publicly display, publicly perform,
distribute and sublicense the Contribution of such Contributor, if any, and such
derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the
Licensed Patents. The patent license shall not apply to any other combinations
which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses
to its Contributions set forth herein, no assurances are provided by any
Contributor that the Program does not infringe the patent or other intellectual
property rights of any other entity. Each Contributor disclaims any liability to
Recipient for claims brought by any other entity based on infringement of
intellectual property rights or otherwise. As a condition to exercising the
rights and licenses granted hereunder, each Recipient hereby assumes sole
responsibility to secure any other intellectual property rights needed, if any.
For example, if a third party patent license is required to allow Recipient to
distribute the Program, it is Recipient's responsibility to acquire that license
before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who
includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year
after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

   the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work; moral rights retained by the original author(s) and/or performer(s); publicity and privacy rights pertaining to a person's image or likeness depicted in a Work; rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below; rights protecting the extraction, dissemination, use and reuse of data in a Work; database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.
2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law. Affirmer disclaims responsibility for clearing rights of other persons that may
apply to the Work or any use thereof, including without limitation any person's
Copyright and Related Rights in the Work. Further, Affirmer disclaims
responsibility for obtaining any necessary consents, permissions or other
rights required for any use of the Work. Affirmer understands and acknowledges
that Creative Commons is not a party to this document and has no duty or
obligation with respect to this CC0 or use of the Work.

OpenIB.org BSD license (FreeBSD Variant)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Default Dual License

Unless otherwise stated this software is available to you under a choice of
one of two licenses. You may choose to be licensed under the terms of the
OpenIB.org BSD (MIT variant) license (see COPYING.BSD_MIT) or the GNU General
Public License (GPL) Version 2 (see COPYING.GPL2), both included in this
package.

Files marked 'See COPYING file' are licensed under the above Dual License.

# Other Options

Individual source files may use a license different from the above Default Dual
License. If a license is declared in the file then it supersedes the Default
License.

If a directory contains a COPYING file then the License from that file becomes
the Default License for files in that directory and below.

# Copyright Holders
Refer to individual files for information on the copyright holders.

# License Catalog (Informative, Non Binding)

## Utilities

Utility source code that may be linked into any binary are available under several licenses:

- MIT license (see ccan/LICENSE.MIT)
- Creative Commons CC0 1.0 Universal License (see ccan/LICENSE.CC0)

## Providers

The following providers use a different license than the Default Dual License. Refer to files in each directory for details.

- hfi1verbs
  : Dual License: GPLv2 or Intel 3 clause BSD license

- ipathverbs
  : Dual License: GPLv2 or PathScale BSD Patent license

- ocrdma
  : Dual License: GPLv2 or OpenIB.org BSD (FreeBSD variant), See COPYING.BSD_FB

## Libraries

All library compilable source code (.c and .h files) are available under the Default Dual License.

Unmarked ancillary files may be available under a Dual License: GPLv2 or OpenIB.org BSD (FreeBSD variant).

## Tools (iwpmd, srp_daemon, ibacm)

All compilable source code (.c and .h files) are available under the Default Dual License.

Unmarked ancillary files may be available under a Dual License: GPLv2 or OpenIB.org BSD (FreeBSD variant).

srp_daemon/srp_daemon/srp_daemon.sh: Any one of the GPLv2, a 2 clause BSD license or the CPLv1.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.146 flex-json 2.1
1.146.1 Available under license:
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.147 spring-beans 4.3.22.RELEASE
1.147.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
* /

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/BeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/MapFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/ObjectFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/BeanDefinitionBuilder.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/SetFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/CannotLoadBeanClassException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/BeanDefinitionDefaults.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/parsing/Problem.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/CustomScopeConfigurer.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/GenericTypeAwareAutowireCandidateResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/parsing/BeanComponentDefinition.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/PropertyAccessException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/UnsatisfiedDependencyException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/BeanCreationException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/NamedBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/FactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/InstantiationAwareBeanPostProcessorAdapter.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/NamedBeanHolder.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/YamlMapFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/YamlPropertiesFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/NoSuchBeanDefinitionException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilNamespaceHandler.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilTypeConverterSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilXmlTypeConverter.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilXmlTypeConverterSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesTypeConverterSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesTypeConverterSupportSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/UtilYamlPropertiesTypeConverterSupportSupportSupport.java
*/
Copyright 2002-2019 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/support/AbstractBeanDefinition.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/config/PlaceholderConfigurerSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/support/ReplaceOverride.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/config/ConstructorArgumentValues.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/propertyeditors/PathEditor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/config/ReplacingBeanFactoryPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/support/BeanDefinitionParserDelegate.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/support/BeanDefinitionAdvisorSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3.jar/org/springframework/beans/factory/config/ReplaceAdviceFactorySupport.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/PropertyOverrideConfigurer.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/groovy/GroovyDynamicElementReader.groovy
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/BeanDefinitionReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/BeanConfigurerSupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/access/BeanFactoryReference.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/AbstractBeanDefinitionReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/AutowireCandidateResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/PropertiesBeanDefinitionReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/BeanInfoFactory.java
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/propertyeditors/TimeZonEditor.java
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/config/TypedStringValue.java
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/ExtendedBeanInfoFactory.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2006 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License”);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/propertyeditors/ByteArrayPropertyEditor.java
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/parsing/BeanEntry.java
*/
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/propertyeditors/CharArrayPropertyEditor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License”);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/XmlBeanDefinitionReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/Aware.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/BeanWrapperImpl.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/AutowiredAnnotationBeanPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/AbstractFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/DestructionAwareBeanPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/DependencyDescriptor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/AbstractAutowireCapableBeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/DefaultNamespaceHandlerResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/MutablePropertyValues.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/support/PagedListHolder.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/support/AbstractBeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/TypeMismatchException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/groovy/GroovyBeanDefinitionReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/groovy/GroovyBeanDefinitionWrapper.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/PluggableSchemaResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/AnnotationBeanWiringInfoResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/RequiredAnnotationBeanPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/groovy/GroovyBeanDefinitionWrapper.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/TypeMismatchException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/groovy/GroovyBeanDefinitionReader.java
jar/org/springframework/beans/DirectFieldAccessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/InitializingBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/annotation/Autowired.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/annotation/QualifierAnnotationAutowireCandidateResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/propertyeditors/URIEditor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/BeanFactoryUtils.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/ListableBeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/support/BeanDefinitionRegistry.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/support/StaticListableBeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/support/FactoryBeanRegistrySupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/support/DefaultSingletonBeanRegistry.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/DisposeableBean.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2008 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/PropertyAccessorFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/config/BeanExpressionResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/serviceloader/ServiceListFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/access/el/SimpleSpringBeanELResolver.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2002-2010 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
jar/org/springframework/beans/factory/support/BeanDefinitionRegistryPostProcessor.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/BeanPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/BeanDefinitionDocumentReader.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/InitDestroyAnnotationBeanPostProcessor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/SingletonBeanRegistry.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/ConfigurableListableBeanFactory.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/xml/BeansDtdResolver.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/PropertyEditorRegistrySupport.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/Lookup.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/AnnotationBeanUtils.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/NotReadablePropertyException.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2014 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/NotReadablePropertyException.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
*limitations under the License.*
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/config/BeanExpressionContext.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/annotation/Autowire.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2002-2007 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/BeanNameGenerator.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/serviceloader/AbstractServiceLoaderBasedFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/PropertyValuesEditor.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/serviceloader/ServiceFactoryBean.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/parsing/ReaderEventListener.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2002-2012 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/DefaultBeanNameGenerator.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/ManagedArray.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/factory/support/SortDefinition.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/support/ManagedProperties.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/support/SortDefinition.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/support/MutableSortDefinition.java
* /opt/cola/permits/1141041592_1614924789.21/0/spring-beans-4-3-22-release-sources-3-jar/org/springframework/beans/propertyeditors/ClassArrayEditor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
1.148 simpleclient-common 0.8.1
1.148.1 Available under license :
   Apache-2.0

1.149 jdk-zulu 8.50.0.51
1.149.1 Available under license :
   The copyrights in this software and any visual or audio work distributed with the software belong to Azul Systems, Inc. and those included in all other notice files either listed in the readme file or contained in any other included notice files with this distribution. All rights are reserved. Installation of this software and any Azul software bundled with or derived from this software is licensed only in accordance with these terms.

   Provided you have not received the software directly from Azul and have already agreed to the terms of a separate license agreement, by installing, using or distributing this software you, on your own behalf and on behalf of your employer or principal, agree to be bound by these terms. If you do not agree to any of these terms, you may not use, copy, transmit, distribute nor install this software.

   The software is developed and owned by Azul and/or any of its affiliates, subsidiaries or respective suppliers and licensors. The software also includes certain software components and materials ("Open Source Materials") of third parties ("Third Party Licensors") licensed under certain licenses ("Open Source Licenses").

   Nothing in this Agreement limits your rights under, or grants you rights that supersede the terms and conditions of any applicable Open Source Licenses. You must review the Open Source Licenses located at http://www.azul.com/license/zulu_third_party_licenses.html to understand your rights under them. In the event that no such third party license agreements exist, the restrictions contained in this Agreement shall...
With respect to the Open Source Licenses the following shall apply:

1) You hereby acknowledge and agree that you will be licensing any Open Source Materials directly from the applicable Third Party Licensors including the right to use such Third Party Materials in connection with the software.

2) In the event of any inconsistencies or conflicting provisions of the Open Source Licenses and the provisions of this Agreement, the provisions of the Open Source Licenses shall prevail.

THIS SOFTWARE IS PROVIDED BY AZUL "AS-IS" AND WITHOUT EXPRESS OR IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL AZUL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES (INCLUDING BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICE, LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY WHETHER IN CONTRACT, STRICT LIABILITY OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the
Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program
or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body
of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible
use to the public, the best way to achieve this is to make it free software
which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach
them to the start of each source file to most effectively convey the exclusion
of warranty; and each file should have at least the "copyright" line and a
pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by the Free
Software Foundation; either version 2 of the License, or (at your option)
any later version.

This program is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for
more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it
starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes
with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free
software, and you are welcome to redistribute it under certain conditions;
type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may be
called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school,
if any, to sign a "copyright disclaimer" for the program, if necessary. Here
is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.150 commons-beanutils 1.8.3

1.150.1 Available under license:

Apache Commons BeanUtils
Copyright 2000-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 1326
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.151 netty-resolver-dns 4.1.49.Final

1.151.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsNameResolverException.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/DnsServerAddressStream.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/SequentialDnsServerAddressStream.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/RotationalDnsServerAddresses.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/ShuffledDnsServerAddressStream.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/SingletonDnsServerAddresses.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *  *
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/DnsCache.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/InflightNameResolver.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-jar/io/netty/resolver/dns/NoopDnsCache.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DnsAddressResolveContext.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DnsCnameCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DefaultAuthoritativeDnsServerCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DnsAddressDecoder.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/Cache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DefaultDnsCnameCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/AuthoritativeDnsServerCacheAdapter.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/NameServerComparator.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/NoopAuthoritativeDnsServerCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/NoopDnsCnameCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DnsRecordResolveContext.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/AuthoritativeDnsServerCache.java
  * /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
    jar/io/netty/resolver/dns/DirContextUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 */
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsServerAddresses.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsAddressResolverGroup.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsNameResolver.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/package-info.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsResolveContext.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/DnsQueryContext.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2017 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/SingletonDnsServerAddressStreamProvider.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
jar/io/netty/resolver/dns/TraceDnsQueryLifecycleObserver.java
* /opt/ws_local/PERMITS_SQL/1068720131_1594427102.23/0/netty-resolver-dns-4-1-49-final-sources-1-
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Sysvinit is Copyright (C) 1991-2004 Miquel van Smoorenburg

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2`.

Send patches to sysvinit-devel@nongnu.org

The of the start-stop-daemon

* A rewrite of the original Debian's start-stop-daemon Perl script
* in C (faster - it is executed many times during system startup).
* Written by Marek Michalkiewicz <marekm@i17linuxb.ists.pwr.wroc.pl>,
* public domain.

### 1.153 shiro-cache 1.7.1

#### 1.153.1 Available under license:

Apache Shiro

Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based on initial ideas from Dr. Heinz Kabutz's publicly posted version available at http://www.javaspecialists.eu/archive/Issue015.html, with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.154 base-passwd 3.5.44

1.154.1 Available under license:

Name: base-passwd
Maintainer: Colin Watson <cjwatson@debian.org>

Files: update-passwd.c, man/*
Copyright: Copyright 1999-2002 Wichert Akkerman <wichert@deephackmode.org>
Copyright 2002, 2003, 2004 Colin Watson <cjwatson@debian.org>
License: GPL-2

Files: passwd.master, group.master
License: PD
X-Notes: Originally written by Ian Murdock <imurdock@debian.org> and Bruce Perens <bruce@pixar.com>.

Files: doc/*
Copyright: Copyright 2001, 2002 Joey Hess
Copyright 2007 David Mandelberg
License: GPL-2

License: GPL-2
On Debian and Debian-based systems, a copy of the GNU General Public License version 2 is available in /usr/share/common-licenses/GPL-2.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in any medium for any purpose).

The "Program" need not be published under this License unless it also contains some notice that it is available under this License. However, once you have received or downloaded such a program, you can choose to distribute it under this License, with or without modifications, or you canirty or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in any medium for any purpose).
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.155 commons-collections 3.2.2
1.155.1 Available under license :
Apache Commons Collections
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work.
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.156 netty-buffer 4.1.49.Final

1.156.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/io/netty/buffer/PooledHeapByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
 */

Found in path(s):
*/opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/io/netty/buffer/WrappedCompositeByteBuf.java
No license file was found, but licenses were detected in source scan.

~/Copyright 2012 The Netty Project

~/The Netty Project licenses this file to you under the Apache License,
~/version 2.0 (the "License"); you may not use this file except in compliance
~/with the License. You may obtain a copy of the License at:
~/http://www.apache.org/licenses/LICENSE-2.0

~/Unless required by applicable law or agreed to in writing, software
~/distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~/WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~/License for the specific language governing permissions and limitations
~/under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/META-INF/maven/io.netty/netty-buffer/pom.xml

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
jar/io/netty/buffer/PooledByteBufAllocatorMetric.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
jar/io/netty/buffer/ByteBufAllocatorMetricProvider.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
jar/io/netty/buffer/ByteBufAllocatorMetric.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2020 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
jar/io/netty/buffer/search/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PoolChunkList.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PoolChunk.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PooledDirectByteBuf.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/ReadOnlyByteBuf.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PoolArena.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
  jar/io/netty/buffer/UnsafeDirectSwappedByteBuf.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-
  jar/io/netty/buffer/UnsafeHeapSwappedByteBuf.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/AbstractUnsafeSwappedByteBuf.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PoolChunkMetric.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4.1-49-final-sources-1.jar/io/netty/buffer/PooledUnsafeHeapByteBuf.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2020 The Netty Project
 *
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* /

Found in path(s):
No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/META-INF/native-image/io.netty/buffer/native-image.properties

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/io/netty/buffer/EmptyByteBuf.java
* /opt/ws_local/PERMITS_SQL/1068232274_1594324101.89/0/netty-buffer-4-1-49-final-sources-1-jar/io/netty/buffer/FixedCompositeByteBuf.java
1.157 vertx-web 3.9.4

1.157.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Red Hat, Inc.
 *
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 *
 * The Eclipse License is available at
 *
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 *
 * You may elect to redistribute this code under either of these licenses.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 Red Hat, Inc.
 *
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 *
 * The Eclipse Public License is available at
 *
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 *
 * You may elect to redistribute this code under either of these licenses.
 */

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/RouteState.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/MultiTenantHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/RouterState.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/MultiTenantHandler.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2017 Red Hat, Inc.
 *
 * Red Hat licenses this file to you under the Apache License, version 2.0
 * (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
*/
*
* Copyright 2015 Red Hat, Inc.
*
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
*
* The Eclipse Public License is available at
*
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
*
* You may elect to redistribute this code under either of these licenses.
*/

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-
jar/io/vertx/ext/web/handler/impl/CSRFHandlerImpl.java
No license file was found, but licenses were detected in source scan.

*/
*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/cookie.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2014 Red Hat, Inc.
*
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
*
* The Eclipse Public License is available at
*
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
*
* You may elect to redistribute this code under either of these licenses.
*/
/
* Copyright (c) 2011-2013 The original author or authors
* .................................................................
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
*
* The Eclipse Public License is available at
*
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
*
* You may elect to redistribute this code under either of these licenses.
*/

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/SockJSHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/XhrTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/EventSourceTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/BaseTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/SockJSSocketBase.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/SockJSSession.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/EventBusBridgeImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/TransportListener.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/
Open Source Used In IOx-FogDirector-AC18 1.17.0 1370

/*
 * Copyright 2014 Red Hat, Inc.
 *
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 *
 * The Eclipse Public License is available at
 *
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 *
 * You may elect to redistribute this code under either of these licenses.
 */

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/HtmlFileTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/JsonCodec.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/SockJSSocket.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/WebSocketTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/RawWebSocketTransport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/Transport.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/impl/JsonPTransport.java

*/

*/

*/
jar/io/vertx/ext/web/handler/LoggerHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/RouterImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/UserSessionHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/SockJSBridgeOptions.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/FileUploadImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/AuthHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/FaviconHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/RoutingContextImplBase.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/CorsHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/sstore/impl/ClusteredSessionStoreImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/Error_Handler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/Error_HandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/FormLoginHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/OAuth2AuthHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/BodyHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/impl/CookieImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/OAuth2AuthHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/Session.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/RedirectAuthHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/CookieHandlerImpl.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/TemplateHandler.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/sockjs/SockJSHandlerOptions.java
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-jar/io/vertx/ext/web/handler/impl/LoggerHandlerImpl.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Red Hat, Inc.
 *
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 *
 * The Eclipse Public License is available at
 *
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 *
 * You may elect to redistribute this code under either of these licenses.
 */

Found in path(s):
* /opt/cola/permits/1101064165_1603114191.51/0/vertx-web-3-9-4-sources-
1.158 jandex 2.1.3.Final
1.158.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * JBoss, Home of Professional Open Source.
 * Copyright 2014 Red Hat, Inc., and individual contributors
 * as indicated by the @author tags.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *  http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/ParameterizedType.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/Utils.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/FieldInfoGenerator.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/MethodInfoGenerator.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/ThrowsTypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/TypeParameterBoundTypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/EmptyTypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/PositionBasedTypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/TypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/AnnotationTargetFilterCollection.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/PositionBasedTypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/TypeTarget.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
jar/org/jboss/jandex/AnnotationTargetFilterCollection.java
* /opt/ws_local/PERMITS_SQL/1063930911_1593103343.43/0/jandex-2-1-3-final-sources-
No license file was found, but licenses were detected in source scan.

/*
 * JBoss, Home of Professional Open Source.
 * Copyright 2013 Red Hat, Inc., and individual contributors
 * as indicated by the @author tags.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
1.159 guava 29.0-jre

1.159.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2010 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Equivalence.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/ContiguousSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/UncaughtExceptionHandlers.java
jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/ListeningExecutorService.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/SortedLists.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/Atomics.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/annotations/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/primitives/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/annotations/Beta.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/**
 * Holder for extra methods of {code Objects} only in web. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/base/ExtraObjectsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingDeque.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/TransformedIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/FilteredKeySetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingNavigableSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/TransformedListIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingBlockingDeque.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingImmutableList.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/AbstractMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/CompactHashSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/FilteredKeyListMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/RangeMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/SortedMultisetBridge.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/FilteredMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/TreeRangeMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/RegularImmutableAsList.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/DescenderingImmutableSortedSet.java
*/
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

No license file was found, but licenses were detected in source scan.

Found in path(s):
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2005 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/reflect/Reflection.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

/**
 * Holder for web specializations of methods of [@code Ints]. Intended to be empty for regular
 * version.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/primitives/IntsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*/
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of [code Shorts]. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/primitives/ShortsMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collection/BaseImmutableMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collection/JdkBackedImmutableMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collection/JdkBackedImmutableBiMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collection/IndexedImmutableSet.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2007 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/DirectExecutor.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/eventbus/Subscribe.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Function.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Defaults.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/Files.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/eventbus/EventBus.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/Flushables.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/CountingOutputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/FinalizableSoftReference.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ExecutionList.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/CompactHashing.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/IncidentEdgeSet.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Outer class that exists solely to let us write [ @code Partially.GwtIncompatible ] instead of plain
* [ @code GwtIncompatible ]. This is more accurate for [ @link Futures#catching ], which is available
* under GWT but with a slightly different signature.
*
* <p>We can't use [ @code PartiallyGwtIncompatible ] because then the GWT compiler wouldn't recognize
* it as a [ @code GwtIncompatible ] annotation. And for [ @code Futures.catching ], we need the GWT
* compiler to autostrip the normal server method in order to expose the special, inherited GWT
* version.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/Partially.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright (C) 2014 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/base/MoreObjects.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/eventbus/Dispatcher.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/ListenerCallQueue.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/math/Quantiles.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/eventbus/Subscriber.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/eventbus/SubscriberRegistry.java

No license file was found, but licenses were detected in source scan.

/
* Copyright (C) 2008 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/ImmutableMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/StandardTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/HashBasedTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/Collections2.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/ImmutableListMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/RegularImmutableBiMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/ImmutableMapEntrySet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/UnmodifiableIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/ImmutableCollection.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/Serialization.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/EmptyImmutableListMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/PeekingIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/ImmutableMapValues.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/Table.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableEntry.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableMapKeySet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableSortedSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/CollectPreconditions.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableBiMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/Tables.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/Range.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/StandardRowSortedTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/SingletonImmutableBiMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/TreeBasedTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/Platform.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/RegularImmutableMap.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/CollectSpliterators.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/ImmutableBiMapFauxverideShim.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/package-info.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/PatternCompiler.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/CommonPattern.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/JdkPattern.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/CommonMatcher.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */
This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group
and released to the public domain, as explained at
http://creativecommons.org/licenses/publicdomain

As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's
java.util.HashMap class.

Copyright (C) 2011 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (C) 2015 The Guava Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/ReaderInputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/CombinedFuture.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/AggregateFutureState.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/AsyncCallable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/ConsumingQueueIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/MacHashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/LittleEndianByteArray.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/InterruptibleTask.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/Platform.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/FarmHashFingerprint64.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/LittleEndianByteArray.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2020 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
/*
/**
 * Holder for web specializations of methods of [ @code Doubles }. Intended to be empty for regular
 * version.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/hash/package-info.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/ArrayTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/ForwardingTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License is
distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

No license file was found, but licenses were detected in source scan.
/*
  * Copyright (C) 2017 The Guava Authors
  *
  * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
  * in compliance with the License. You may obtain a copy of the License at
  *
  * http://www.apache.org/licenses/LICENSE-2.0
  *
  * Unless required by applicable law or agreed to in writing, software distributed under the License
  * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
  * express
  * or implied. See the License for the specific language governing permissions and limitations under
  * the License.
  */

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/ImmutableLongArray.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/AbstractHashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ForwardingCondition.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1110737159_1606846951.59/jar/com/google/common/primitives/Platform.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/
/**
* Holder for web specializations of methods of [code Floats]. Intended to be empty for regular
* version.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/jar/com/google/common/primitives/FloatsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/jar/com/google/common/primitives/Platform.java

* /opt/cola/permits/1110737159_1606846951.59/jar/com/google/common/util/concurrent/Internal.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2015 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you
 * may not use this file except in compliance with the License. You may
 * obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
 * implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/Streams.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2013 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/ImmutableMapEntry.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/io/MoreFiles.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collect/MultimapBuilder.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2013 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/io/CharSequenceReader.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/Runnables.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/FilteredMultimapValues.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/WrappingScheduledExecutorService.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/base/Utf8.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/AbstractTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/base/Verify.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/eventbus/SubscriberExceptionContext.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/*
* This following method is a modified version of one found in
* http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30
* which contained the following notice:
* 
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
* the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/
* 
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/MoreExecutors.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2012 The Guava Authors
* 
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
* */
/*
* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
* following header:
* 
* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
* hereby disclaims copyright to this source code.
* */
Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/base/SmallCharMatcher.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright (C) 2007 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/  
/**  
* Returns an array containing all of the elements in the specified collection. This method  
* returns the elements in the order they are returned by the collection's iterator. The returned  
* array is "safe" in that no references to it are maintained by the collection. The caller is  
* thus free to modify the returned array.  
*  
* <p>This method assumes that the collection size doesn't change while the method is running.  
*  
* <p>TODOR(kevinb): support concurrently modified collections?  
*  
* @param c the collection for which to return an array of elements  
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collection/ObjectArrays.java
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright (C) 2010 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/MinMaxPriorityQueue.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/SortedMapDifference.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/RowSortedTable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingImmutableCollection.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/AbstractSequentialIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/UnmodifiableListIterator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingSetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingListMultimap.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/StandardMutableValueGraph.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/StandardMutableValueGraph.java
Open Source Used In IOx-FogDirector-AC18 1.17.0 1403

jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/MutableValueGraph.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/ImmutableMultisetGwtSerializationDependencies.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/AbstractGraphBuilder.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/CollectCollectors.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/GraphBuilder.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/EndpointPair.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/ArrayListMultimapGwtSerializationDependencies.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
/*
 * Copyright (C) 2012 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 */

No license file was found, but licenses were detected in source scan.
Open Source Used In IOx-FogDirector-AC18 1.17.0 1405

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/math/StatsAccumulator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/reflect/Element.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/math/PairedStatsAccumulator.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/hash/SipHashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/io/CharSink.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/html/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/cache/LongAddable.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/reflect/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/math/LinearTransformation.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/common/collect/CartesianList.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/FilteredKeyMultimap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableRangeMap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/CharSource.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/Closer.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ServiceManager.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/PairedStats.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/xml/package-info.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/Stats.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/Network.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/RecursiveDeleteOption.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/PredecessorsFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/TopKSelector.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/ImmutableNetwork.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/Graph.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/MutableGraph.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/Graphs.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright (C) 2009 The Guava Authors
* *
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
/*
 * Copyright 2019 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/IgnoreJRERequirement.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2011 The Guava Authors
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/cache/RemovalListener.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/AbstractSortedMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/HashingOutputStream.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/UnsignedLongs.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/hash/PrimitiveSink.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/base/Present.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/collection/EmptyContiguousSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/base/ Enums.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/hash/Murmur3_128HashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/cache/RemovalCause.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/net/HttpHeaders.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/base/Ticker.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/util/concurrent/ForwardingListeningExecutorService.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/cache/ForwardingLoadingCache.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3.jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/base/Absent.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/ForwardingExecutorService.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/DoubleMath.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/primitives/UnsignedInteger.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/cache/ForwardingCache.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/cache/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collect/RegularImmutableMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/TreeRangeSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/AbstractHasher.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/math/package-info.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/hash/AbstractStreamingHashFunction.java
No license file was found, but licenses were detected in source scan.
/*
 * Copyright (C) 2007 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/RegularImmutableSet.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ExplicitOrdering.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/AbstractSetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/AbstractMapBasedMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingSortedMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/HashMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ClassToInstanceMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/Iterators.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/LinkedHashMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingConcurrentMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ForwardingObject.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/TreeMultiset.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/MapDifference.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/UsingToStringOrdering.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/SetMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/NaturalOrdering.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/NullsLastOrdering.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/ForwardingQueue.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/AbstractMapBasedMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/ForwardingMultimap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/Multiset.java
No license file was found, but licenses were detected in source scan.

/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/hash/LongAdder.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/hash/Striped64.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/cache/Striped64.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/cache/LongAdder.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/AtomicDoubleArray.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2006 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 The Guava Authors
*/
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/Traverser.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/graph/AbstractBaseGraph.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the
 * License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-jar/com/google/common/collection/ImmutableSortedAsList.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/cache/CacheBuilder.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/collection/MapMaker.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/ForwardingFuture.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/net/InternetDomainName.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/html/HtmlEscapers.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/cache/ReferenceEntry.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/base/Platform.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/ForwardingFluentFuture.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/annotations/GwtIncompatible.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/io/ByteProcessor.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/Cut.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/MapMakerInternalMap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/MapMakerInternalMap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/base/Splitter.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/Service.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/annotations/GwtCompatible.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/cache/LocalCache.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/DenseImmutableTable.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/MapMakerInternalMap.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/SpliceEscapers.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/ForwardingListenableFuture.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/util/concurrent/AbstractIdleService.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/SparseImmutableTable.java
* /opt/cola/permits/110737159_1606846951.59/0/guava-29-0-jre-sources-3-
jar/com/google/common/collect/RegularImmutableTable.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/ServiceManagerBridge.java

1.160 spring-web-mvc 4.3.22.RELEASE

1.160.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2002-2015 the original author or authors.  
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1110737159_1606846951.59/0/guava-29-0-jre-sources-3-
  jar/com/google/common/util/concurrent/ServiceManagerBridge.java

* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/resource/DefaultResourceTransformerChain.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/JasperReportsMultiFormatView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/script/ScriptTemplateConfig.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/FlashMap.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/support/RequestContextUtils.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/condition/RequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/ServletWrappingController.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/method/annotation/JsonViewResponseBodyAdvice.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/handler/MatchableHandlerMapping.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/condition/AbstractMediaTypeExpression.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/handler/HandlerInterceptorAdapter.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/view/groovy/GroovyMarkupView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/AsyncHandlerInterceptor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/method/annotation/MatrixVariableMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/resource/AbstractVersionStrategy.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/resource/CssLinkResourceTransformer.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/condition/AbstractRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/view/groovy/GroovyMarkupView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/AsyncHandlerInterceptor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/mvc/method/annotation/MatrixVariableMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * A contract for inspecting and potentially modifying request data values such
 * as URL query parameters or form field values before they are rendered by a
 * view or before a redirect.
 *
 * <p>Implementations may use this contract for example as part of a solution
 * to provide data integrity, confidentiality, protection against cross-site
 * request forgery (CSRF), and others or for other tasks such as automatically
 * adding a hidden field to all forms and URLs.
 *
 * <p>View technologies that support this contract can obtain an instance to
 * delegate to via <tt>{@link RequestContext#getRequestDataValueProcessor()}</tt>.
 *
 * @author Rossen Stoyanchev
Open Source Used In IOx-FogDirector-AC18 1.17.0 1427

* @since 3.1
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/support/RequestDataValueProcessor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2008 the original author or authors.
*Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
*You may obtain a copy of the License at
*http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
*distributed under the License is distributed on an "AS IS" BASIS,
*WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*See the License for the specific language governing permissions and
*limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2.jar/org/springframework/web/servlet/mvc/support/ControllerTypePredicate.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2016 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
*You may obtain a copy of the License at
*http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
*distributed under the License is distributed on an "AS IS" BASIS,
*WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*See the License for the specific language governing permissions and
*limitations under the License.
*/

/**
* XSLT-driven View that allows for response context to be rendered as the
*result of an XSLT transformation.
*
The XSLT Source object is supplied as a parameter in the model and then
[@link #locateSource detected] during response rendering. Users can either specify
a specific entry in the model via the [@link #setSourceKey sourceKey] property or
have Spring locate the Source object. This class also provides basic conversion
of objects into Source implementations. See [@link #getSourceTypes() here]
for more details.

All model parameters are passed to the XSLT Transformer as parameters.
In addition the user can configure [@link #setOutputProperties output properties]
to be passed to the Transformer.

@author Rob Harrop
@author Juergen Hoeller
@since 2.0
*/

/*
 * Copyright 2002-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/support/AbstractAnnotationConfigDispatcherServletInitializer.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/config/ViewControllerBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/condition/HeadersRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/AbstractUrlBasedView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/i18n/CookieLocaleResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/resource/WebJarsResourceResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/support/AbstractFlashMapManager.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/freemarker/FreeMarkerView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/annotation/ResourceHandlerRegistration.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/script/ScriptTemplateView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ServletModelAttributeMethodProcessor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/resource/ResourceUrlProvider.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/resource/ResourceTransformerSupport.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/EscapeBodyTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ResponseBodyEmitterReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/annotation/InterceptorRegistry.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/ViewResolversBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ExtendedServletRequestDataBinder.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/annotation/AsyncSupportConfigurer.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/condition/ParamsRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/support/RedirectAttributesModelMap.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/groovy/GroovyMarkupConfigurer.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ResponseBodyAdvice.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/annotation/ContentNegotiationConfigurer.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/HandlerExecutionChain.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/Param.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/annotation/PathMatchConfigurer.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ModelAndViewMethodReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/DeferredResultMethodReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/AbstractMessageConverterMethodProcessor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/velocity/VelocityViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/velocity/VelocityLayoutView.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0

 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/ViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/velocity/VelocityViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/handler/AbstractDetectingUrlHandlerMapping.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/EditorAwareTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/velocity/VelocityLayoutView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
Open Source Used In IOx-FogDirector-AC18 1.17.0 1433
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2010 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License"; you may not use this file except in compliance with the License. You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/velocity/VelocityToolboxView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/handler/DispatcherServletWebRequest.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2018 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/condition/ProducesRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/handler/MappedInterceptor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/LabelTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/UrlTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/condition/ConsumesRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/ScriptTemplateConfigurerBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/condition/CompositeRequestCondition.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
Open Source Used In IOx-FogDirector-AC18 1.17.0  1436

jar/org/springframework/web/servlet/view/json/AbstractJackson2View.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/AnnotationDrivenBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/resource/VersionResourceResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/FormTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/ResponseBodyEmitter.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/support/BindStatus.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/AbstractDataBoundFormElementTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/SelectTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/FrameworkServlet.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/method/annotation/MvcUriComponentsBuilder.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/FreeMarkerConfigurerBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/mvc/support/DefaultHandlerExceptionResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/CheckboxTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/InputTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/CheckboxesTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/view/BeanNameViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/tags/form/CheckboxTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/servlet/config/FreeMarkerConfigurerBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/resource/PathResourceResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/mvc/method/annotation/ExceptionHandlerExceptionResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/tags/form/TagWriter.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/handler/HandlerMappingIntrospector.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/config/TilesConfigurerBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/View.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/resource/ResourceHttpRequestHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/UrlBasedViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/tiles2/SpringBeanPreparerFactory.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/freemarker/FreeMarkerViewResolver.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/tiles2/SpringBeanPreparerFactory.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/freemarker/FreeMarkerViewResolver.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/tiles2/SpringLocaleResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/support/AbstractControllerUrlHandlerMapping.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/feed/AbstractFeedView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/PathVariableMapMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/feed/AbstractAtomFeedView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/JasperReportsCsvView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/condition/RequestConditionHolder.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/PathVariableMapMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/JasperReportsXlsView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/tiles3/SpringWildcardServletTilesApplicationContext.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/RedirectAttributesMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/JasperReportsResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/UrlFilenameViewController.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/ViewMethodReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/AbstractJasperReportsSingleFormatView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/tiles2/AbstractTilesApplicationContext.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/RedirectAttributesMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/AbstractJasperReportsXlsView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/ViewMethodReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/RedirectAttributesMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/ViewMethodReturnValueHandler.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/method/annotation/RedirectAttributesMethodArgumentResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/jasperreports/AbstractJasperReportsXlsView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/mvc/UrlFilenameViewController.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-jar/org/springframework/web/mvc/method/annotation/ServletRequestDataBinderFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/theme/SessionThemeResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/support/JstlUtils.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/velocity/VelocityView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/ThemeResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/i18n/AbstractLocaleContextResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/LocaleContextResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/theme/CookieThemeResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/ThemeTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/i18n/AbstractLocaleResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/document/AbstractJExcelView.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/BindErrorsTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/form/AbstractHtmlInputElementTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/form/AbstractFormTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/BindTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/HandlerAdapter.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/view/AbstractCachingViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/config/InterceptorsBeanDefinitionParser.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/tags/tags/HtmlEscapeTag.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/theme/ThemeChangeInterceptor.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
  jar/org/springframework/web/servlet/theme/FixedThemeResolver.java
jar/org/springframework/web/servlet/tags/form/HiddenInputTag.java
No license file was found, but licenses were detected in source scan.

/*/ 
* Copyright 2002-2007 the original author or authors. 
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software 
* distributed under the License is distributed on an "AS IS" BASIS, 
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
* See the License for the specific language governing permissions and 
* limitations under the License. 
*/

Found in path(s):
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/view/AbstractTemplateViewResolver.java
* /opt/cola/permits/1141298140_1614968794.16/0/spring-webmvc-4-3-22-release-sources-2-
jar/org/springframework/web/servlet/theme/AbstractThemeResolver.java

1.161 diffutils 3.6-1
1.161.1 Available under license:
This is the Debian prepackaged version of the GNU diffutils package.
GNU `diff` was written by Mike Haertel, David Hayes, Richard Stallman,
Len Tower, and Paul Eggert. Wayne Davison designed and implemented
the unified output format. GNU `diff3` was written by Randy Smith.
GNU `sdiff` was written by Thomas Lord. GNU `cmp` was written by
Torbjorn Granlund and David MacKenzie.

The source for this release was obtained from


Program copyright and license:

===============================================

Free Software Foundation, Inc.

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

On Debian systems, the complete text of the GNU General Public License
may be found in `/usr/share/common-licenses/GPL'.
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/config/AspectComponentDefinition.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/SpringProxy.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/config/AspectEntry.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/BeforeAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/AopInfrastructureBean.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/config/AbstractInterceptorDrivenBeanDefinitionDecorator.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/aopalliance/intercept/MethodInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/aopalliance/intercept/ConstructorInterceptor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/ControlFlowPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/PerformanceMonitorInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/JamonPerformanceMonitorInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/AspectJAdviceParameterNameDiscoverer.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/AbstractMonitoringInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/AbstractTraceInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/ComposablePointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/Scope/ScopedProxyUtils.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/annotation/AnnotationMatchingPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/autoproxy/AbstractAdvisorAutoProxyCreator.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/TypePatternClassFilter.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/ProxyFactoryBean.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/NameMatchMethodPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
/*<
* Copyright 2002-2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
  jar/org/springframework/aop/config/SimpleBeanFactoryAwareAspectInstanceFactory.java

No license file was found, but licenses were detected in source scan.

/*/  
* Copyright 2002-2014 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/RegexpMethodPointcutAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/target/PrototypeTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/autoproxy/target/LazyInitTargetSourceCreator.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/aop/AdvisorChainFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/target/ThreadLocalTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/AsyncUncaughtExceptionHandler.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/NameMatchMethodPointcutAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/InstantiationModelAwarePointcutAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/AspectJPrecedenceInformation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/interceptor/ExposeInvocationInterceptor.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2006 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/InstantiationModelAwarePointcutAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/AspectJPrecedenceInformation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/support/ExpressionPointcut.java

No license file was found, but licenses were detected in source scan.
/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/autoproxy/AbstractBeanFactoryAwareAdvisingPostProcessor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/ReflectiveMethodInvocation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Advised.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/DefaultAopProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/autoproxy/AutoProxyUtils.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/TargetClassAware.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/AspectJMethodBeforeAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/aspectj/AspectJAroundAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/AbstractSingletonProxyFactoryBean.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractPoolingTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractSingletonProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractPoolingTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractSingletonProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractPoolingTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractSingletonProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractPoolingTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractSingletonProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/framework/Target/AbstractPoolingTargetSource.java
jar/org/springframework/aop/aspectj/AspectJAfterAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/aspectj/AspectJAfterThrowingAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/TrueMethodMatcher.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/aspectj/autoproxy/AspectJPrecedenceComparator.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/framework/ObjenesisCglibAopProxy.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/MethodMatchers.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/aspectj/annotation/PrototypeAspectInstanceFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/framework/ObjectMethodInvocation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/Pointcuts.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * */

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/aspectj/RuntimeTestWalker.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/aspectj/AspectJWeaverMessageHandler.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/framework/MethodInvocation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/framework/SupportInvocation.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/ClassFilters.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/ClassFilter.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/annotation/AnnotationClassFilter.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-
jar/org/springframework/aop/support/annotation/AnnotationMethodMatcher.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 1451
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/config/AopConfigUtils.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/support/AbstractRegexpMethodPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/aspectj/DeclareParentsAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/target/AbstractPrototypeBasedTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/config/AopNamespaceUtils.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopAdvisingBeanPostProcessor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/aspectj/AbstractAspectJAdvice.java
/
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/AfterReturningAdvice.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/IntroductionInfo.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/config/PointcutEntry.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/interceptor/DebugInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/AopInvocationException.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/config/ScopedProxyBeanDefinitionDecorator.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopConfigException.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopConfigAdvisor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/target/LazyInitTargetSource.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopProxyFactory.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/support/DelegatingIntroductionInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopProxy.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/support/AspectJProxyUtils.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/support/DynamicMethodMatcherPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/AopProxy.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/framework/adapter/GlobalAdvisorAdapterRegistry.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/support/DelegatePerTargetObjectIntroductionInterceptor.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3.jar/org/springframework/aop/support/JdkRegexpMethodPointcut.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/target/PoolingConfig.java
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/target/dynamic/Refreshable.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2008 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297475_1614969018.02/0/spring-aop-4-3-22-release-sources-3-jar/org/springframework/aop/ThrowsAdvice.java

1.163 netty-transport 4.1.49.Final
1.163.1 Available under license :
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/group/ChannelMatcher.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/MessageSizeEstimator.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ConnectTimeoutException.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/nio/SelectedSelectionKeySet.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/socket/oio/OioServerSocketChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/socket/oio/OioSocketChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/DefaultMessageSizeEstimator.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/DefaultChannelProgressivePromise.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/bootstrap/ChannelFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/group/DefaultChannelGroup.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelProgressiveFuture.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/socket/oio/DefaultOioServerSocketChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/group/ChannelMatchers.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/SimpleChannelInboundHandler.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/group/ChannelGroupException.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/DefaultChannelId.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelGroup.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/AddressedEnvelope.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelProgressiveFutureListener.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/AddressedEnvelope.java
jar/io/netty/channel/ChannelProgressivePromise.java
*/
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelOutboundBuffer.java
*/
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelId.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
/**
* Handles an I/O event or intercepts an I/O operation, and forwards it to its next handler in
* its [@link ChannelPipeline].
*
* [@link ChannelHandler] itself does not provide many methods, but you usually have to implement one of its
* subtypes:
* <ul>
* <li>[@link ChannelInboundHandler] to handle inbound I/O events, and</li>
* <li>[@link ChannelOutboundHandler] to handle outbound I/O operations.</li>
* </ul>
* </p>
* <p>
* Alternatively, the following adapter classes are provided for your convenience:
* <ul>
* <li>[@link ChannelInboundHandlerAdapter] to handle inbound I/O events,</li>
* <li>[@link ChannelOutboundHandlerAdapter] to handle outbound I/O operations, and</li>
* <li>[@link ChannelDuplexHandler] to handle both inbound and outbound events</li>
* </ul>
* </p>
* <p>
* For more information, please refer to the documentation of each subtype.
* </p>
* 
* <h3>The context object</h3>
A `ChannelHandler` is provided with a `ChannelHandlerContext` object. A `ChannelHandler` is supposed to interact with the `ChannelPipeline` it belongs to via a context object. Using the context object, the `ChannelHandler` can pass events upstream or downstream, modify the pipeline dynamically, or store the information (using `AttributeKey`)s which is specific to the handler.

### State management

A `ChannelHandler` often needs to store some stateful information. The simplest and recommended approach is to use member variables:

```java
public interface Message {
    // your methods here
}

public class DataServerHandler extends SimpleChannelInboundHandler<Message> {
    private boolean loggedIn;

    @Override
    public void channelRead0(ChannelHandlerContext ctx, Message message) {
        if (message instanceof LoginMessage) {
            authenticate((LoginMessage) message);
            loggedIn = true;
        } else (message instanceof GetDataMessage) {
            if (loggedIn) {
                ctx.writeAndFlush(fetchSecret((GetDataMessage) message));
            } else {
                fail();
            }
        }
    }

    // Create a new handler instance per channel.
    // See [ChannelInitializer#initChannel(Channel)].
    public class DataServerInitializer extends ChannelInitializer<Channel> {
        @Override
        public void initChannel(Channel channel) {
            channel.pipeline().addLast("handler", new DataServerHandler());
        }
    }
}
```

Because the handler instance has a state variable which is dedicated to one connection, you have to create a new handler instance for each new channel to avoid a race condition where a unauthenticated client can get the confidential information.
* } }
* }
* }
* </pre>
* 
* <h4>Using <code>@link AttributeKey</code>s</h4>
* 
* Although it's recommended to use member variables to store the state of a
* handler, for some reason you might not want to create many handler instances.
* In such a case, you can use <code>@link AttributeKey</code>s which is provided by
* <code>@link ChannelHandlerContext</code>:
* 
* public interface Message {
*     // your methods here
* }
* 
* { @code @Sharable }
* public class DataServerHandler extends <code>@link SimpleChannelInboundHandler</code>&lt;Message&gt; { 
*     private final <code>@link AttributeKey</code>&lt;@link Boolean&gt; auth =
*         <code>@link AttributeKey</code>#valueOf(String) AttributeKey.valueOf("auth");
* 
* { @code @Override }
*     public void channelRead(<code>@link ChannelHandlerContext</code> ctx, Message message) {
*         <code>@link Attribute</code>&lt;@link Boolean&gt; attr = ctx.attr(auth);
*         if (message instanceof LoginMessage) {
*             authenticate((LoginMessage) o);
*             &lt;b&gt; attr.set(true)&lt;/b&gt; ;
*         } else (message instanceof GetDataMessage) {
*             if (&lt;b&gt;Boolean.TRUE.equals(attr.get())&lt;/b&gt;) {
*                 ctx.writeAndFlush(fetchSecret((GetDataMessage) o));
*             } else {
*                 fail();
*             }
*         }
*         ...
* } }
* </pre>
* Now that the state of the handler is attached to the <code>@link ChannelHandlerContext</code>, you can add the
* same handler instance to different pipelines:
* 
* public class DataServerInitializer extends <code>@link ChannelInitializer</code>&lt;@link Channel&gt; { 
*     private static final DataServerHandler <b>SHARED</b> = new DataServerHandler();
* 
* { @code @Override }
*     public void initChannel(<code>@link Channel</code> channel) {
*         channel.pipeline().addLast("handler", <b>SHARED</b>);
* }
The `@Sharable` annotation

In the example above which used an `AttributeKey`, you might have noticed the `@Sharable` annotation.

If a `ChannelHandler` is annotated with the `@Sharable` annotation, it means you can create an instance of the handler just once and add it to one or more `ChannelPipeline`'s multiple times without a race condition.

If this annotation is not specified, you have to create a new handler instance every time you add it to a pipeline because it has unshared state such as member variables.

This annotation is provided for documentation purpose, just like the JCIP annotations.

Additional resources worth reading

Please refer to the `ChannelHandler`, and `ChannelPipeline` to find out more about inbound and outbound operations, what fundamental differences they have, how they flow in a pipeline, and how to handle the operation in your application.

Found in path(s):

```
/netty-transport-4-1-49-final-sources-1.jar/io/netty/channel/ChannelHandler.java
```

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2018 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 */
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1.jar/io/netty/channel/socket/nio/NioChannelOption.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1.jar/io/netty/channel/SimpleUserEventChannelHandler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1.jar/io/netty/channel/CoalescingBufferQueue.java
No license file was found, but licenses were detected in source scan.

;/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version 2.0 (the
 * "License"); you may not use this file except in compliance with the License. You may obtain a
 * copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/AbstractCoalescingBufferQueue.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelOutboundInvoker.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/SelectStrategy.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/SelectStrategyFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/SelectStrategyFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/ChannelPoolHandler.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/FixedChannelPool.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/ChannelHealthChecker.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel(MaxBytesRecvByteBufAllocator.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/SimpleChannelPool.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/ChannelPool.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/ChannelPoolMap.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/pool/DefaultMaxBytesRecvByteBufAllocator.java
* No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelException.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ThreadPerChannelEventLoopGroup.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/embedded/EmbeddedSocketAddress.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/socket/oio/OioSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/socket/nio/NioDatagramChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/FailedChannelFuture.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelDuplexHandler.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/nio/NioEventLoop.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelPromiseNotifier.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/group/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/group/DefaultChannelGroupFuture.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelOutboundHandler.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelOption.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/socket/DatagramChannelConfig.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/embedded/EmbeddedChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/local/LocalServerChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelPromiseAggregator.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelOutboundHandlerAdapter.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelPipeline.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/socket/oio/OioServerSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/socket/ServerSocketChannel.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/ChannelInboundHandler.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/local/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/socket/DatagramPacket.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/group/ChannelGroupFuture.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/nio/package-info.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-
jar/io/netty/channel/nio/package-info.java
*/

No license file was found, but licenses were detected in source scan.

/*
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
 *
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
Open Source Used In IOx-FogDirector-AC18 1.17.0 1473

*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/EventLoopTaskQueueFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ExtendedClosedChannelException.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelHandlerMask.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ReflectiveChannelFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/embedded/EmbeddedChannelId.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/ChannelFactory.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/PendingWriteQueue.java
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/io/netty/channel/AbstractEventLoop.java

No license file was found, but licenses were detected in source scan.

# The Netty Project licenses this file to you under the Apache License,
# version 2.0 (the "License"); you may not use this file except in compliance
# with the License. You may obtain a copy of the License at:
# # http://www.apache.org/licenses/LICENSE-2.0
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232313_1594323914.29/0/netty-transport-4-1-49-final-sources-1-jar/META-
1.164 fast-infoset 1.2.15

1.165 sed 4.4 2

1.165.1 Available under license:

@c The GNU Free Documentation License.
center Version 1.3, 3 November 2008

c This file is intended to be included within another document,
c hence no sectioning command or @node.

display
uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
display

enumerate 0
item
PREAMBLE

The purpose of this License is to make a manual, textbook, or other
functional and useful document @dfn{free} in the sense of freedom: to
assure everyone the effective freedom to copy and redistribute it,
with or without modifying it, either commercially or noncommercially.
Secondarily, this License preserves for the author and publisher a way
to get credit for their work, while not being considered responsible
for modifications made by others.

This License is a kind of ``copyleft'', which means that derivative
works of the document must themselves be free in the same sense. It
complements the GNU General Public License, which is a copyleft
license designed for free software.

We have designed this License in order to use it for manuals for free
software, because free software needs free documentation: a free
program should come with manuals providing the same freedoms that the
software does. But this License is not limited to software manuals;
it can be used for any textual work, regardless of subject matter or
whether it is published as a printed book. We recommend this License
principally for works whose purpose is instruction or reference.
APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or
for automatic translation to a variety of formats suitable for input
to text formatters. A copy made in an otherwise Transparent file
format whose markup, or absence of markup, has been arranged to thwart
or discourage subsequent modification by readers is not Transparent.
An image format is not Transparent if used for any substantial amount
of text. A copy that is not ``Transparent'' is called "Opaque".

Examples of suitable formats for Transparent copies include plain
ASCII without markup, Texinfo input format, LaTeX input
format, SGML or XML using a publicly available
DTD, and standard-conforming simple HTML,
PostScript or PDF designed for human modification. Examples
of transparent image formats include PNG, XCF and
JPG@. Opaque formats include proprietary formats that can be
read and edited only by proprietary word processors, SGML or
XML for which the DTD and/or processing tools are
not generally available, and the machine-generated HTML,
PostScript or PDF produced by some word processors for
output purposes only.

The ``Title Page'' means, for a printed book, the title page itself,
plus such following pages as are needed to hold, legibly, the material
this License requires to appear in the title page. For works in
formats which do not have any title page as such, ``Title Page'' means
the text near the most prominent appearance of the work's title,
preceding the beginning of the body of the text.

The `'publisher'" means any person or entity that distributes copies
of the Document to the public.

A section `Entitled XYZ'" means a named subunit of the Document whose
title either is precisely XYZ or contains XYZ in parentheses following
text that translates XYZ in another language. (Here XYZ stands for a
specific section name mentioned below, such as `Acknowledgements',
'Dedications', `Endorsements', or `History'.) To `Preserve the Title'
of such a section when you modify the Document means that it remains a
section `Entitled XYZ'" according to this definition.

The Document may include Warranty Disclaimers next to the notice which
states that this License applies to the Document. These Warranty
Disclaimers are considered to be included by reference in this
License, but only as regards disclaiming warranties: any other
implication that these Warranty Disclaimers may have is void and has
no effect on the meaning of this License.

@item
VERBATIM COPYING
You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

@item
COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the
Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A

Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

State on the Title page the name of the publisher of the Modified Version, as the publisher.

Preserve all the copyright notices of the Document.

Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

@item
For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

@item
Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

@item
Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

@item
Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

@item
Preserve any Warranty Disclaimers.
@end enumerate

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the
list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

@item COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."
COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice
or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

@item
TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

@item
FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or
of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

RELICENSING

``Massive Multiauthor Collaboration Site'' (or `"MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

`"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

`Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is `"eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:
to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents.
States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a
menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated
conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the
Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product. A product
is a consumer product regardless of whether the product has substantial
commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source. The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.

If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as
part of a transaction in which the right of possession and use of the
User Product is transferred to the recipient in perpetuity or for a
fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied
by the Installation Information. But this requirement does not apply
if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has
been installed in ROM).

The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates
for a work that has been modified or installed by the recipient, or for
the User Product in which it has been modified or installed. Access to a
network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law. If additional permissions
apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.
If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.
10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent
(such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.
THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

1.166 tdb 1.44.1
1.166.1 Available under license :
GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS
0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities.
However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty
5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for
which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the
Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is
not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w’ and `show c’ should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <https://www.gnu.org/licenses/why-not-lgpl.html>.

1.167 jakarta-validation-api 2.0.2
1.167.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Jakarta Bean Validation API
 *
 * License: Apache License, Version 2.0
 * See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
 */

Found in path(s):
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/Validator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/metadata/CascadableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/ConstraintTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/constraints/FutureOrPresent.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/constraints/Null.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/ConstraintValidatorContext.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/metadata/ExecutableDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/packages/Default.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/MessageInterpolator.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/metadata/package-info.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/valueextraction/Unwrapping.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/valueextraction/ValueExtractor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/constraintvalidation/ValidationTarget.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/metadata/MethodDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/package-info.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/UnexpectedTypeException.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/metadata/BeanDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-jar/java/validation/constraints/Digits.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
Open Source Used In IOx-FogDirector-AC18 1.17.0

jar/javax/validation/metadata/ElementDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/metadata/ContainerElementTypeDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/GroupDefinitionException.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/Min.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/groups/ConvertGroup.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/metadata/Scope.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/Max.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/ReportAsSingleViolation.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/metadata/ReturnValueDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/ParameterNameProvider.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/Pattern.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/NoProviderFoundException.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/Validation.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/ValidatorContext.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/NotBlank.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/spi/package-info.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/Constraint.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/Valid.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/metadata/CrossParameterDescriptor.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/ClockProvider.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/Configuration.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/executable/ValidateOnExecution.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/Size.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
jar/javax/validation/constraints/PastOrPresent.java
* /opt/cola/permits/1135880235_1613624044.2/0/jakarta-validation-api-2-0-2-sources-3-
No license file was found, but licenses were detected in source scan.

~ Jakarta Bean Validation API
~
~ License: Apache License, Version 2.0
~ See the license.txt file in the root directory or <http://www.apache.org/licenses/LICENSE-2.0>.
"[CDATA[
Comments to: <a href="mailto:bean-validation-dev@eclipse.org">bean-validation-dev@eclipse.org</a>.<br>
Copyright © 2019 Eclipse Foundation.<br>
Use is subject to <a href="{@docRoot}/doc-files/speclicense.html" target="_top">EFSL</a>; this spec is based on material that is licensed under the Apache License, version 2.0.]]>

1.168 javax-annotation-api 1.3
1.168.1 Available under license:
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this
License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

   B. Any new file that contains any part of the Original Software or previous Modification; or

   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

   (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

   (b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the
name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF
INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software, (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and commercial computer software documentation, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term)
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable
form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL,
SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

1.169 netty-resolver 4.1.49.Final
1.169.1 Available under license :

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:
~
~   http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io.netty/netty-resolver/pom.xml

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
*   http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/DefaultNameResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/DefaultAddressResolverGroup.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/HostsFileEntriesResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/InetNameResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/CompositeNameResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/HostsFileParser.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/AbstractAddressResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/InetSocketAddressResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/AddressResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-jar/io/netty/resolver/DefaultHostsFileEntriesResolver.java

No license file was found, but licenses were detected in source scan.
/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/ResolvedAddressTypes.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/HostsFileEntries.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/NoopAddressResolverGroup.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/AddressResolverGroup.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/NameResolver.java
* /opt/ws_local/PERMITS_SQL/1068232293_1594320747.73/0/netty-resolver-4-1-49-final-sources-1-
  jar/io/netty/resolver/NoopAddressResolver.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

1.170 metrics-servlets 4.1.9
1.170.1 Available under license :
   Apache-2.0

1.171 ceph 12.2.13-0ubuntu0.18.04.10
1.171.1 Available under license :
   Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
   Upstream-Name: ceph
   Upstream-Contact: Sage Weil <sage@newdream.net>
   Source: http://ceph.com/
   Files-Excluded:
   debian
   src/civetweb/examples/websocket_client/ssl/server.key.orig
   src/civetweb/resources/cert/client.key.orig
   src/civetweb/resources/cert/server.key.orig
   src/erasure-code/jerasure/jerasure/Examples/makefile.orig
   src/erasure-code/jerasure/jerasure/include/config.h.in~
   src/erasure-code/jerasure/jerasure/makefile.orig
Files: *
Copyright: 2004-2014 Sage Weil <sage@newdream.net>
   2004-2014 Inktank <info@inktank.com>
      Inktank, Inc
         Inktank Storage, Inc.
   2012-2014 Red Hat <contact@redhat.com>
   2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
   2013 CohortFS, LLC
   2004-2011 Dreamhost
   2013 eNovance SAS <licensing@enovance.com>
   2014 Adam Crume <adamcrume@gmail.com>
   2012 Florian Haas, hastexo
   2010 Greg Farnum <gregf@hq.newdream.net>
   2014 John Spray <john.spray@inktank.com
   2004-2012 New Dream Network
   2014 Sebastien Ponce <sebastien.ponce@cern.ch>
   2011 Stanislav Sedov <stas@FreeBSD.org>
   2013-2014 UnitedStack <haomai@unitedstack.com>
   2011 Wido den Hollander <wido@widodh.nl>
License: LGPL-2.1

Files: src/erasure-code/jerasure/ErasureCode*
   src/erasure-code/ErasureCode*
   src/erasure-code/isa/*
   src/include/str_map.h
   src/test/common/test_str_map.cc
   src/test/erasure-code/*
   src/test/rgw/test_rgw_manifest.cc
Copyright: 2014 CERN/Switzerland
   2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
   2014 Red Hat <contact@redhat.com>
   2013 eNovance SAS <licensing@enovance.com>
License: LGPL-2.1+

Files: src/erasure-code/isa/isa-l/erasure_code/*
Copyright: 2011-2014 Intel Corporation
License: BSD-3-clause

Files: src/rocksdb/*
Copyright: 2004-2013 Facebook, Inc.
   2011 The LevelDB Authors
   2009 Google Inc.
License: BSD-3-clause
Comment:
   Additional Grant of Patent Rights
.
Software means the rocksdb software distributed by Facebook, Inc.
Facebook hereby grants you a perpetual, worldwide, royalty-free, non-exclusive, irrevocable (subject to the termination provision below) license under any rights in any patent claims owned by Facebook, to make, have made, use, sell, offer to sell, import, and otherwise transfer the Software. For avoidance of doubt, no license is granted under Facebook's rights in any patent claims that are infringed by (i) modifications to the Software made by you or a third party, or (ii) the Software in combination with any software or other technology provided by you or a third party.

The license granted hereunder will terminate, automatically and without notice, for anyone that makes any claim (including by filing any lawsuit, assertion or other action) alleging (a) direct, indirect, or contributory infringement or inducement to infringe any patent: (i) by Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, (ii) by any party if such claim arises in whole or in part from any software, product or service of Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, or (iii) by any party relating to the Software; or (b) that any right in any patent claim of Facebook is invalid or unenforceable.

Files: src/rocksdb/util/xxhash.*
Copyright: 2012-2014, Yann Collet.
License: BSD-2-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: src/mount/canonicalize.c
Copyright: 1993 Rick Sladkey <jrs@world.std.com>
2013 Inktank <info@inktank.com>
2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
License: LGPL-2+

Files: src/os/btrfs_ioctl.h
src/test/mon/PGMap.cc
Copyright: 2007 Oracle. All rights reserved.
2014 Inktank <info@inktank.com>
License: GPL-2

Files: src/common/ceph_hash.cc
License: public-domain
This file uses Robert Jenkin's hash function as detailed at:

http://burtleburtle.net/bob/hash/evahash.html

This is in the public domain.

Files: src/common/bloom_filter.hpp
Copyright: 2000 Arash Partow
License: Boost-Software-License-1.0
Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement,
including the above license grant, this restriction and the following
disclaimer, must be included in all copies of the Software, in whole or
in part, and all derivative works of the Software, unless such copies
or derivative works are solely in the form of machine-executable object
code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Files: src/common/crc32c_intel*
Copyright: 2012-2013 Intel Corporation All Rights Reserved.
License: BSD-3-clause

Files: src/common/sctp_crc32.c
Copyright: 2001-2007, by Cisco Systems, Inc. All rights reserved, 2004-2006 Intel Corporation - All Rights Reserved
License: BSD-3-clause

Files: src/erasure-code/jerasure/gf-complete/*/*
Copyright: 2013 James S. Plank
   Ethan L. Miller
   Kevin M. Greenan
   Benjamin A. Arnold
   John A. Burnum
   Adam W. Disney
   Allen C. McBride
License: BSD-3-clause
Comment:
https://bitbucket.org/jimplank/gf-complete

Files: src/erasure-code/jerasure/jerasure/*/*
Copyright: 2011-2013 James S. Plank <plank@cs.utk.edu> 2013 Kevin Greenan
License: BSD-3-clause

Files: src/gtest/*
Copyright: 2008, Google Inc.
License: BSD-3-clause

Files: src/civetweb/*
Copyright: 2004-2013 Sergey Lyubka 2013-2014 the Civetweb developers
License: Expat

Files: src/java/native/ScopedLocalRef.h src/java/native/JniConstants.*
Copyright: 2010 The Android Open Source Project
License: Apache-2.0
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The complete text of the Apache License, Version 2.0
can be found in "/usr/share/common-licenses/Apache-2.0".

Files: src/libs3/*
Copyright: 2008 Bryan Ischo <bryan@ischo.com>
License: GPL-3/OpenSSL
libs3 is free software: you can redistribute it and/or modify it under the
terms of the GNU General Public License as published by the Free Software
Foundation, version 3 of the License.

In addition, as a special exception, the copyright holders give
permission to link the code of this library and its programs with the
OpenSSL library, and distribute linked combinations including the two.

libs3 is distributed in the hope that it will be useful, but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE. See the GNU General Public License for more
details.

The complete text of the GNU General Public License version 3
can be found in "/usr/share/common-licenses/GPL-3" file.

Files: src/mount/mtab.c
Copyright: util-linux-ng AUTHORS
License: GPL-2+
Comment:
"mount/fstab.c" from line 559:
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/mount-deprecated/fstab.c?h=v2.22#n559
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/README.licensing

Files: src/test/librbd/fsx.c
Copyright: 1991, NeXT Computer, Inc.
License: APSL-2.0
The contents of this file constitute Original Code as defined in and
are subject to the Apple Public Source License Version 2.0 (the
"License"). You may not use this file except in compliance with the License. Please obtain a copy of the License at http://www.opensource.apple.com/apsl/ and read it before using this file.

This Original Code and all software distributed under the License are distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. Please see the License for the specific language governing rights and limitations under the License.

Comment:
http://codemonkey.org.uk/projects/fsx/
http://codemonkey.org.uk/projects/fsx/fsx-macosforge/fsx.c

Files: man/*
    debian/man/*

Copyright: 2010-2014, Inktank Storage, Inc. and contributors.
License: CC-BY-SA-3.0

Files: debian/*

Copyright: 2010      Sage Weil <sage@newdream.net>
    2010      Canonical, Ltd.
    2011-2013 Lszl Bszrmnyi (GCS) <gcs@debian.org>
    2013-2014 James Page <james.page@ubuntu.com>
    2014      Dmitry Smirnov <onlyjob@debian.org>

License: LGPL-2.1

License: GPL-2
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2` file.

License: GPL-2+
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 2 of the License, or
(at your option) any later version.

This package is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU General Public License
version 2 can be found in `/usr/share/common-licenses/GPL-2`.

License: LGPL-2.1
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License version 2.1 as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU Lesser General
Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

License: LGPL-2.1+
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU Lesser General
Public License can be found in `/usr/share/common-licenses/LGPL-2.1’.

License: LGPL-2+
This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License version 2 (or later) as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU Lesser General
Public License 2 can be found in `/usr/share/common-licenses/LGPL-2’.

License: Expat
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Comment:
This license also known as "MIT" however FSF consider "MIT" labelling
ambiguous and copyright-format specification recommend to label such license
as "Expat".

License: CC-BY-SA-3.0
Creative Commons Attribution-ShareAlike 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE
LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN
ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION
ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE 
INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM 
ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE 
COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY 
COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS 
AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE 
TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY 
BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS 
CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND 
CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and 
other pre-existing works, such as a translation, adaptation, derivative 
work, arrangement of music or other alterations of a literary or 
artistic work, or phonogram or performance and includes cinematographic 
adaptations or any other form in which the Work may be recast, 
transformed, or adapted including in any form recognizably derived from 
the original, except that a work that constitutes a Collection will not 
be considered an Adaptation for the purpose of this License. For the 
avoidance of doubt, where the Work is a musical work, performance or 
phonogram, the synchronization of the Work in timed-relation with a 
moving image ("synching") will be considered an Adaptation for the 
purpose of this License.

b. "Collection" means a collection of literary or artistic works, such 
as encyclopedias and anthologies, or performances, phonograms or 
broadcasts, or other works or subject matter other than works listed in 
Section 1(f) below, which, by reason of the selection and arrangement of 
their contents, constitute intellectual creations, in which the Work is 
included in its entirety in unmodified form along with one or more other 
contributions, each constituting separate and independent works in 
themselves, which together are assembled into a collective whole. A work 
that constitutes a Collection will not be considered an Adaptation (as 
defined below) for the purposes of this License.

c. "Creative Commons Compatible License" means a license that is listed 
at http://creativecommons.org/compatiblelicenses that has been approved 
by Creative Commons as being essentially equivalent to this License, 
including, at a minimum, because that license: (i) contains terms that 
have the same purpose, meaning and effect as the License Elements of
this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.

d. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

e. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.

f. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

g. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

h. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

i. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the
Licensor to exercise rights under this License despite a previous violation.

j. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

k. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in
which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.
b. You may Distribute or Publicly Perform an Adaptation only under the
terms of: (i) this License; (ii) a later version of this License with
the same License Elements as this License; (iii) a Creative Commons
jurisdiction license (either this or a later license version) that
contains the same License Elements as this License (e.g.,
Attribution-ShareAlike 3.0 US)); (iv) a Creative Commons Compatible
License. If you license the Adaptation under one of the licenses
mentioned in (iv), you must comply with the terms of that license. If
you license the Adaptation under the terms of any of the licenses
mentioned in (i), (ii) or (iii) (the "Applicable License"), you must
comply with the terms of the Applicable License generally and the
following provisions: (I) You must include a copy of, or the URI for,
the Applicable License with every copy of each Adaptation You Distribute
or Publicly Perform; (II) You may not offer or impose any terms on the
Adaptation that restrict the terms of the Applicable License or the
ability of the recipient of the Adaptation to exercise the rights
granted to that recipient under the terms of the Applicable License;
(III) You must keep intact all notices that refer to the Applicable
License and to the disclaimer of warranties with every copy of the Work
as included in the Adaptation You Distribute or Publicly Perform; (IV)
when You Distribute or Publicly Perform the Adaptation, You may not
impose any effective technological measures on the Adaptation that
restrict the ability of a recipient of the Adaptation from You to
exercise the rights granted to that recipient under the terms of the
Applicable License. This Section 4(b) applies to the Adaptation as
incorporated in a Collection, but this does not require the Collection
apart from the Adaptation itself to be made subject to the terms of the
Applicable License.

c. If You Distribute, or Publicly Perform the Work or any Adaptations or
Collections, You must, unless a request has been made pursuant to
Section 4(a), keep intact all copyright notices for the Work and
provide, reasonable to the medium or means You are utilizing: (i) the
name of the Original Author (or pseudonym, if applicable) if supplied,
and/or if the Original Author and/or Licensor designate another party or
parties (e.g., a sponsor institute, publishing entity, journal) for
attribution ("Attribution Parties") in Licensor's copyright notice,
terms of service or by other reasonable means, the name of such party or
parties; (ii) the title of the Work if supplied; (iii) to the extent
reasonably practicable, the URI, if any, that Licensor specifies to be
associated with the Work, unless such URI does not refer to the
copyright notice or licensing information for the Work; and (iv) ,
consistent with Section 3(b), in the case of an Adaptation, a credit
identifying the use of the Work in the Adaptation (e.g., "French
translation of the Work by Original Author," or "Screenplay based on
original Work by Original Author"). The credit required by this Section
4(c) may be implemented in any reasonable manner; provided, however,
that in the case of a Adaptation or Collection, at a minimum such credit
will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors.

For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

d. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License.
Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on
September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License.

Creative Commons may be contacted at http://creativecommons.org/.

License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials
provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.172 classmate 1.5.1

1.172.1 Available under license:

Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel
This copy of Java ClassMate library is licensed under Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.173 cpp 8.4.0 1ubuntu1~18.04

1.173.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices"
to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any
non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to
produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified
it, and giving a relevant date.

b) The work must carry prominent notices stating that it is
released under this License and any conditions added under section
7. This requirement modifies the requirement in section 4 to
"keep intact all notices".

c) You must license the entire work, as a whole, under this
License to anyone who comes into possession of a copy. This
License will therefore apply, along with any applicable section 7
additional terms, to the whole of the work, and all its parts,
regardless of how they are packaged. This License gives no
permission to license the work in any other way, but it does not
invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display
Appropriate Legal Notices; however, if the Program has interactive
interfaces that do not display Appropriate Legal Notices, your
work need not make them do so.

A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.
A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.
"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further
restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission
to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free
patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.
Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.
Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read

1.174 gnutls 3.5.18-1ubuntu1.5
1.174.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it:

responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict
development and use of software on general-purpose computers, but in those that do, we wish to avoid the special
danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures
that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”.
“Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission,
other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a
work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or
secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a
private copy. Propagation includes copying, distribution (with or without modification), making available to the
public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere
interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and
prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no
warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under
this License, and how to view a copy of this License. If the interface presents a list of user commands or options,
such as a menu, a prominent item in the list meets this criterion.


The “source code” for a work means the preferred form of the work for making modifications to it. “Object code”
means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body,
or, in the case of interfaces specified for a particular programming language, one that is widely used among
developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included
in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves
only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating
that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all
notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty
protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of
source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
b) The work must carry prominent notices stating that it is released under this License and any conditions added
under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy.
   This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work,
   and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any
   other way, but it does not invalidate such permission if you have separately received it.
d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program
   has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature
extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a
volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright
are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit.
Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also
convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium),
   accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software
   interchange.
b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium),
   accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or
   customer support for that product model, to give anyone who possesses the object code either (1) a copy of the
   Corresponding Source for all the software in the product that is covered by this License, on a durable physical
   medium customarily used for software interchange, for a price no more than your reasonable cost of physically
   performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no
   charge.
c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source.
   This alternative is allowed only occasionally and noncommercially, and only if you received the object code with
   such an offer, in accord with subsection 6b.
d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though
they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies
you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.
In the following three paragraphs, a “patent license” is any express agreement or commitment, however
denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for
patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitmen
to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is
not available for anyone to copy, free of charge and under the terms of this License, through a publicly available
network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so
available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3)
arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream
recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying
the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more
identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring
conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work
authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license
you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of,
or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License.
You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of
distributing software, under which you make payment to the third party based on the extent of your activity of
conveying the work, and under which the third party grants, to any of the parties who would receive the covered
work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you
(or copies made from those copies), or (b) primarily for and in connection with specific products or compilations
that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to
infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of
this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as
to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a
consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty
for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms
and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with
a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to
convey the resulting work. The terms of this License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through
a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <https://www.gnu.org/licenses/why-not-lgpl.html>.
1.175 jetty-jndi 9.4.31.v20200723

1.175.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.
2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, perform derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html
and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------
OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.176 latencyutils 2.0.3

1.177 debianutils 4.8.4

1.177.1 Available under license:

This is the Debian GNU/Linux package debianutils.

It is an original Debian package. Programs in it were maintained by Guy Maor <maor@debian.org>, and are now maintained by Clint Adams <schizo@debian.org>.

All its programs except savelog, and which may be redistributed under the terms of the GNU GPL, Version 2 or later, found on Debian systems in the file /usr/share/common-licenses/GPL, which is in the public domain.

savelog may be redistributed under the following terms: (The rest of this file consists of savelog's distribution terms.)

#ident"@(#)smail:RELEASE-3_2:COPYING,v 1.2 1996/06/14 18:59:10 woods Exp"

SMAIL GENERAL PUBLIC LICENSE
(Clarified 11 Feb 1988)

Copyright (C) 1988 Landon Curt Noll & Ronald S. Karr
Copyright (C) 1992 Ronald S. Karr
Copyleft (GNU) 1988 Landon Curt Noll & Ronald S. Karr

Everyone is permitted to copy and distribute verbatim copies of this license, but changing it is not allowed. You can also use this wording to make the terms for other programs.

The license agreements of most software companies keep you at the mercy of those companies. By contrast, our general public license is
intended to give everyone the right to share SMAIL. To make sure that you get the rights we want you to have, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. Hence this license agreement.

Specifically, we want to make sure that you have the right to give away copies of SMAIL, that you receive source code or else can get it if you want it, that you can change SMAIL or use pieces of it in new free programs, and that you know you can do these things.

To make sure that everyone has such rights, we have to forbid you to deprive anyone else of these rights. For example, if you distribute copies of SMAIL, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

Also, for our own protection, we must make certain that everyone finds out that there is no warranty for SMAIL. If SMAIL is modified by someone else and passed on, we want its recipients to know that what they have is not what we distributed, so that any problems introduced by others will not reflect on our reputation.

Therefore we (Landon Curt Noll and Ronald S. Karr) make the following terms which say what you must do to be allowed to distribute or change SMAIL.

COPYING POLICIES

1. You may copy and distribute verbatim copies of SMAIL source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy a valid copyright notice "Copyright (C) 1988 Landon Curt Noll & Ronald S. Karr" (or with whatever year is appropriate); keep intact the notices on all files that refer to this License Agreement and to the absence of any warranty; and give any other recipients of the SMAIL program a copy of this License Agreement along with the program. You may charge a distribution fee for the physical act of transferring a copy.

2. You may modify your copy or copies of SMAIL or any portion of it, and copy and distribute such modifications under the terms of Paragraph 1 above, provided that you also do the following:

   a) cause the modified files to carry prominent notices stating that you changed the files and the date of any change; and

   b) cause the whole of any work that you distribute or publish, that in whole or in part contains or is a derivative of SMAIL or
any part thereof, to be licensed at no charge to all third parties on terms identical to those contained in this License Agreement (except that you may choose to grant more extensive warranty protection to some or all third parties, at your option).

c) You may charge a distribution fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another unrelated program with this program (or its derivative) on a volume of a storage or distribution medium does not bring the other program under the scope of these terms.

3. You may copy and distribute SMAIL (or a portion or derivative of it, under Paragraph 2) in object code or executable form under the terms of Paragraphs 1 and 2 above provided that you also do one of the following:

a) accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Paragraphs 1 and 2 above; or,

b) accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal shipping charge) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Paragraphs 1 and 2 above; or,

c) accompany it with the information you received as to where the corresponding source code may be obtained. (This alternative is allowed only for non-commercial distribution and only if you received the program in object code or executable form alone.)

For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs.

4. You may not copy, sublicense, distribute or transfer SMAIL except as expressly provided under this License Agreement. Any attempt otherwise to copy, sublicense, distribute or transfer SMAIL is void and your rights to use the program under this License agreement shall be automatically terminated. However, parties who have received computer software programs from you with this License Agreement will not have their licenses terminated so long as such parties remain in full compliance.

5. If you wish to incorporate parts of SMAIL into other free programs whose distribution conditions are different, write to Landon Curt Noll & Ronald S. Karr via the Free Software Foundation at 51
Franklin St, Fifth Floor, Boston, MA 02110-1301, USA. We have not yet worked out a simple rule that can be stated here, but we will often permit this. We will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software.

Your comments and suggestions about our licensing policies and our software are welcome! This contract was based on the contract made by the Free Software Foundation. Please contact the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA, or call (617) 542-5942 for details on copylefted material in general.

NO WARRANTY

BECAUSE SMAIL IS LICENSED FREE OF CHARGE, WE PROVIDE ABSOLUTELY NO WARRANTY, TO THE EXTENT PERMITTED BY APPLICABLE STATE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING, LANDON CURT NOLL & RONALD S. KARR AND/OR OTHER PARTIES PROVIDE SMAIL "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF SMAIL IS WITH YOU. SHOULD SMAIL PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW WILL LANDON CURT NOLL & RONALD S. KARR AND/OR ANY OTHER PARTY WHO MAY MODIFY AND REDISTRIBUTE SMAIL AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY LOST PROFITS, LOST MONIES, OR OTHER SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS) SMAIL, EVEN IF YOU HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY.

1.178 glib 2.56.4-0ubuntu0.18.04.9

1.178.1 Available under license:

This work may be reproduced and distributed in whole or in part, in any medium, physical or electronic, so as long as this copyright notice remains intact and unchanged on all copies. Commercial redistribution is permitted and encouraged, but you may not redistribute, in whole or in part, under terms more restrictive than those under which you received it. If you redistribute a modified or translated version of this work, you must also make the source code to the modified or translated version available in electronic form.
without charge. However, mere aggregation as part of a larger work shall not count as a modification for this purpose.

All code examples in this work are placed into the public domain, and may be used, modified and redistributed without restriction.

BECAUSE THIS WORK IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE WORK, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE WORK "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. SHOULD THE WORK PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE WORK AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE WORK, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
That's all there is to it!

1.179 heimdal 7.5.0+dfsg-1

1.179.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@macro copynext{}
@end macro
@vskip 20pt plus 1fil
@end macro

@macro copyrightstart{}
@end macro

@macro copyrightend{}
@end macro

@node Copyrights and Licenses, , Acknowledgments, Top
@comment node-name, next, previous, up
@author
@appendix Copyrights and Licenses

@heading Kungliga Tekniska Hgskolan

@copyrightstart
@verbatim
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@verbatim

Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior
permission. M.I.T. makes no representations about the suitability of
this software for any purpose. It is provided "as is" without express
or implied warranty.

@end verbatim
@copynext

@heading The Regents of the University of California

The parts of the libroken, most of libtelnet, telnet, ftp,
and popper.

@verbatim
Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext
@heading The Regents of the University of California.

libedit

@verbatim

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim

LibTomMath is hereby released into the Public Domain.

@end verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@verbatim

@end verbatim
@end copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim

Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@verbatim

Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim
Copyright (c) 2006,2007 NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
@end verbatim
The NetBSD Foundation, Inc.

vis.c in libbroken

Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

AES in libhcrypto

rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)
This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED ""AS IS"" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@copynext

@heading Richard Outerbridge

DES core in libhcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec’s THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim

@copynext

@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name of the Novell nor the names of its contributors may be used
to endorse or promote products derived from this software without specific
  prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
@verbatim

@copyrightend

1.180 jetty-security 9.4.31.v20200723

1.180.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.
"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under
its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

-----

Oracle OpenJDK
If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.181 audit4j-core 2.6.0
1.181.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.182 vertx-bridge-common 3.9.4
1.182.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.183 curl 7.58.0 2ubuntu3.16

1.183.1 Available under license :
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2020, Daniel Stenberg. <daniel@haxx.se>, and many contributors, see the THANKS file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

1.184 glibc 2.27-3ubuntu1.5

1.184.1 Available under license :

Copyright 1992, 1993, 1994, 1997 Henry Spencer. All rights reserved. This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on any computer system, and to alter it and redistribute it, subject to the following restrictions:

1. The author is not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources,
credits must appear in the documentation.

3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.

4. This notice may not be removed or altered.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.
Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
to loss of data or data being rendered inaccurate or losses sustained by
you or third parties or a failure of the program to operate with any other
programs), even if such holder or other party has been advised of the
possibility of such damages.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

(one line to give the program's name and a brief idea of what it does.)
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program...
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.
For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.
(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/. Unicode Data Files do not include PDF online code charts under the directory http://www.unicode.org/Public/.

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.
COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2013 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

This file contains the copying permission notices for various files in the GNU C Library distribution that have copyright owners other than the Free Software Foundation. These notices all require that a copy of the notice be included in the accompanying documentation and be distributed with binary distributions of the code, so be sure to include this file along with any binary distributions derived from the GNU C Library.

All code incorporated from 4.4 BSD is distributed under the following license:

Copyright (C) 1991 Regents of the University of California.
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. [This condition was removed.]
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The DNS resolver code, taken from BIND 4.9.5, is copyrighted by UC Berkeley, by Digital Equipment Corporation and by Internet Software Consortium. The DEC portions are under the following license:

Portions Copyright (C) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED ``AS IS'' AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
The ISC portions are under the following license:

Portions Copyright (c) 1996-1999 by Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The Sun RPC support (from rpcsrc-4.0) is covered by the following license:

Copyright (c) 2010, Oracle America, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the "Oracle America, Inc." nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The following CMU license covers some of the support code for Mach, derived from Mach 3.0:

Mach Operating System
All Rights Reserved.

Permission to use, copy, modify and distribute this software and its documentation is hereby granted, provided that both the copyright notice and this permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof, and that both notices appear in supporting documentation.

CARNEGIE MELLON ALLOWS FREE USE OF THIS SOFTWARE IN ITS "AS IS" CONDITION. CARNEGIE MELLON DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

Carnegie Mellon requests users of this software to return to

Software Distribution Coordinator
School of Computer Science
Carnegie Mellon University
Pittsburgh PA 15213-3890

or Software.Distribution@CS.CMU.EDU any improvements or extensions that they make and grant Carnegie Mellon the rights to redistribute these changes.

The file if_ppp.h is under the following CMU license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY CARNEGIE MELLON UNIVERSITY AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE UNIVERSITY OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license covers the files from Intel's "Highly Optimized Mathematical Functions for Itanium" collection:

Intel License Agreement

Copyright (c) 2000, Intel Corporation

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name of Intel Corporation may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTEL OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The files inet/getnameinfo.c and sysdeps/posix/getaddrinfo.c are copyright (C) by Craig Metz and are distributed under the following license:
/* The Inner Net License, Version 2.00

The author(s) grant permission for redistribution and use in source and binary forms, with or without modification, of the software and documentation provided that the following conditions are met:

0. If you receive a version of the software that is specifically labelled as not being for redistribution (check the version message and/or README), you are not permitted to redistribute that version of the software in any way or form.
1. All terms of the all other applicable copyrights and licenses must be followed.
2. Redistributions of source code must retain the authors' copyright notice(s), this list of conditions, and the following disclaimer.
3. Redistributions in binary form must reproduce the authors' copyright notice(s), this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
4. [The copyright holder has authorized the removal of this clause.]
5. Neither the name(s) of the author(s) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY ITS AUTHORS AND CONTRIBUTORS `AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

If these license terms cause you a real problem, contact the author. */

The file sunrpc/des_impl.c is copyright Eric Young:

Copyright (C) 1992 Eric Young
Collected from libdes and modified for SECURE RPC by Martin Kuck 1994
This file is distributed under the terms of the GNU Lesser General Public License, version 2.1 or later - see the file COPYING.LIB for details.
If you did not receive a copy of the license with this program, please see <http://www.gnu.org/licenses/> to obtain a copy.

The libidn code is copyright Simon Josefsson, with portions copyright The Internet Society, Tom Tromey and Red Hat, Inc.:


Open Source Used In IOx-FogDirector-AC18 1.17.0 1651
This file is part of GNU Libidn.

GNU Libidn is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

GNU Libidn is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with GNU Libidn; if not, see <http://www.gnu.org/licenses/>.

The following notice applies to portions of libidn/nfkc.c:

This file contains functions from GLIB, including gutf8.c and gunidecomp.c, all licensed under LGPL and copyright hold by:

Copyright (C) 1999, 2000 Tom Tromey
Copyright 2000 Red Hat, Inc.

The following applies to portions of libidn/punycode.c and libidn/punycode.h:

This file is derived from RFC 3492bis written by Adam M. Costello.

Disclaimer and license: Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing...
the copyright notice or references to the Internet Society or other
Internet organizations, except as needed for the purpose of
developing Internet standards in which case the procedures for
copyrights defined in the Internet Standards process must be
followed, or as required to translate it into languages other than
English.

The limited permissions granted above are perpetual and will not be
revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an
"AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

The file inet/rcmd.c is under a UCB copyright and the following:

Copyright (C) 1998 WIDE Project.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the project nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE PROJECT AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE PROJECT OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The file posix/runtests.c is copyright Tom Lord:
Copyright 1995 by Tom Lord

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of the copyright holder not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

Tom Lord DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO
EVENT SHALL TOM LORD BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

The posix/rxspencer tests are copyright Henry Spencer:

Copyright 1992, 1993, 1994, 1997 Henry Spencer. All rights reserved.
This software is not subject to any license of the American Telephone
and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on
any computer system, and to alter it and redistribute it, subject
to the following restrictions:

1. The author is not responsible for the consequences of use of this
software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by
explicit claim or by omission. Since few users ever read sources,
credits must appear in the documentation.

3. Altered versions must be plainly marked as such, and must not be
misrepresented as being the original software. Since few users
ever read sources, credits must appear in the documentation.

4. This notice may not be removed or altered.

The file posix/PCRE.tests is copyright University of Cambridge:

Copyright (c) 1997-2003 University of Cambridge

Permission is granted to anyone to use this software for any purpose on any
computer system, and to redistribute it freely, subject to the following restrictions:

1. This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

2. The origin of this software must not be misrepresented, either by explicit claim or by omission. In practice, this means that if you use PCRE in software that you distribute to others, commercially or otherwise, you must put a sentence like this

   Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England.

   somewhere reasonably visible in your documentation and in any relevant files or online help data or similar. A reference to the ftp site for the source, that is, to

   ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/

   should also be given in the documentation. However, this condition is not intended to apply to whole chains of software. If package A includes PCRE, it must acknowledge it, but if package B is software that includes package A, the condition is not imposed on package B (unless it uses PCRE independently).

3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. If PCRE is embedded in any software that is released under the GNU General Purpose Licence (GPL), or Lesser General Purpose Licence (LGPL), then the terms of that licence shall supersede any condition above with which it is incompatible.

Files from Sun fdlibm are copyright Sun Microsystems, Inc.:

Copyright (C) 1993 by Sun Microsystems, Inc. All rights reserved.

Developed at SunPro, a Sun Microsystems, Inc. business.
Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

Part of stdio-common/tst-printf.c is copyright C E Chew:

(C) Copyright C E Chew
Feel free to copy, use and distribute this software provided:

1. you do not pretend that you wrote it
2. you leave this copyright notice intact.

Various long double libm functions are copyright Stephen L. Moshier:

Copyright 2001 by Stephen L. Moshier <moshier@na-net.ornl.gov>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, see <http://www.gnu.org/licenses/>. */

1.185 netty-transport-native-epoll 4.1.46.Final

1.185.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Set the [@code TCP_MD5SIG] option on the socket. See [@code linux/tcp.h] for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 */
* Allowing them being read would mean anyone with access to the channel could get them.
 */
/**
 * Set the {@code TCP_QUICKACK} option on the socket. See <a href="http://linux.die.net/man/7/tcp">TCP_QUICKACK</a>
 * for more details.
 */

Found in path(s):
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */

Found in path(s):
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java
No license file was found, but licenses were detected in source scan.

/**
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollMode.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollRecvByteAllocatorStreamingHandle.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollChannelConfig.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
Open Source Used In IOx-FogDirector-AC18 1.17.0

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/*
 * Copyright 2014 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 *   http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * No license file was found, but licenses were detected in source scan.
 */

/*
 * No license file was found, but licenses were detected in source scan.
 */
* Copyright 2013 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-
jar/netty_epoll_native.c
* /opt/cola/permits/1283431041_1646722067.76/0/netty-transport-native-epoll-4-1-46-final-sources-
jar/io/netty/channel/epoll/Native.java

1.186 libsigar 1.6.6
1.186.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.187 failureaccess 1.0.1

1.187.1 Available under license :

No license file was found, but licenses were detected in source scan.

    /*
    * Copyright (C) 2018 The Guava Authors
    *
    * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
    * in compliance with the License. You may obtain a copy of the License at
    * http://www.apache.org/licenses/LICENSE-2.0
    */
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/cola/permits/1199742917_1630577017.47/0/failureaccess-1-0-1-sources-3-
  jar/com/google/common/util/concurrent/internal/InternalFutureFailureAccess.java
* /opt/cola/permits/1199742917_1630577017.47/0/failureaccess-1-0-1-sources-3-
  jar/com/google/common/util/concurrent/internal/InternalFutures.java

1.188 jakarta-annotation-api 1.3.5
1.188.1 Available under license :

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from
and are Distributed by that particular Contributor. A Contribution
"originates" from a Contributor if it was added to the Program by
such Contributor itself or anyone acting on such Contributor's behalf.
Contributions do not include changes or additions to the Program that
are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient” means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works” shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works” shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute” means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code” means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License” means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other
combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special,
incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights
in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any
party to be under a license that satisfies the requirements
of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the
Program (i) is combined with other material in a separate file or
files made available under a Secondary License, and (ii) the initial
Contributor attached to the Source Code the notice described in
Exhibit A of this Agreement, then the Program may be made available
under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of
the Program.

3.3 Contributors may not remove or alter any copyright, patent,
trademark, attribution notices, disclaimers of warranty, or limitations
of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add
their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities
with respect to end users, business partners and the like. While this
license is intended to facilitate the commercial use of the Program,
the Contributor who includes the Program in a commercial product
offering should do so in a manner which does not create potential
liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program
in a commercial product offering. The obligations in this section do not
apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified
Contributor must: a) promptly notify the Commercial Contributor in
writing of such claim, and b) allow the Commercial Contributor to control,
and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.
If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

      (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

# Notices for Jakarta Annotations

This content is produced and maintained by the Jakarta Annotations project.

* Project home: https://projects.eclipse.org/projects/ee4j.ca

## Trademarks

Jakarta Annotations is a trademark of the Eclipse Foundation.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU
General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/common-annotations-api

## Third-party Content

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

### 1.189 jboss-transaction-api 1.1.1.Final

### 1.190 commons-pool 1.5.6

#### 1.190.1 Available under license:

Apache Commons Pool
Copyright 2001-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor” shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.191 cdebconf 0.213ubuntu1

1.191.1 Available under license:

CDebConf was initially written by Randolph Chung <tausq@debian.org>

Other contributors include:
Anthony Towns <ajt@debian.org>
David Whedon <dwhedon@gordian.com>
Dan Jacobowitz <dan@debian.org>
Tollef Fog Heen <tfheen@debian.org>
Attilio Fiandrotti <fiandro@tiscali.it>
Colin Watson <cjwatson@debian.org>
    Regis Boudin <regis@debian.org>

CDebConf includes ideas and code from:
debconf - The original, de facto, perl implementation
(c) Joey Hess <joeyh@debian.org>
apt - The Debian Advanced Package Tool
(c) Jason Gunthorpe <jgg@debian.org>
(derived portions are public domain)

CDebConf is copyrighted (c) 2000-2009 by Randolph Chung <tausq@debian.org>,
the d-i team (see above), and Canonical Ltd. under the following license:

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
1.192 vertx-auth-common 3.9.4

1.192.1 Available under license:

No license file was found, but licenses were detected in source scan.

/
* Copyright 2014 Red Hat, Inc.
*
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
*
* The Eclipse Public License is available at
*
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
*
* You may elect to redistribute this code under either of these licenses.
*/

Found in path(s):
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/examples/package-info.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/impl/hash/PBKDF2.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/JWTOptions.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/HashingStrategy.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/ChainAuth.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/HashString.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/HashingAlgorithm.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/AbstractUser.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/examples/AuthCommonExamples.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/PRNG.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/package-info.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/HashingAlgorithm.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/User.java
*/
* Copyright 2015 Red Hat, Inc.
*
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
* 
* The Eclipse Public License is available at
* 
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
* 
* You may elect to redistribute this code under either of these licenses.
*/

Found in path(s):
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/AuthOptions.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/VertxContextPRNG.java
* /opt/cola/permits/1101064199_1603114222.41/0/vertx-auth-common-3-9-4-sources-jar/io/vertx/ext/auth/KeyStoreOptions.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Red Hat, Inc.
*
* All rights reserved. This program and the accompanying materials
* are made available under the terms of the Eclipse Public License v1.0
* and Apache License v2.0 which accompanies this distribution.
*
* The Eclipse Public License is available at
*
* The Apache License v2.0 is available at
* http://www.opensource.org/licenses/apache2.0.php
*
* You may elect to redistribute this code under either of these licenses.
*/

Found in path(s):
1.193 swagger 1.5.20

1.193.1 Available under license:

```java
3nameLjava/lang/String;urlAnnotationDefault
SourceFileLicense.javaRuntimeVisibleAnnotationsLjava/lang/annotation/Target;value"Ljava/lang/annotation/Element
Type;ANNOTATION_TYPE
Ljava/lang/annotation/Retention;&Ljava/lang/annotation/RetentionPolicy;RUNTIMEio/swagger/annotations/License
java/lang/Objectjava/lang/annotation/Annotation&
```
Found in path(s):
* /opt/cola/permits/1280350859_1646202214.38/0/influxdb-java-2-20-sources-jar/org/influxdb/annotation/Measurement.java
* /opt/cola/permits/1280350859_1646202214.38/0/influxdb-java-2-20-sources-jar/org/influxdb/impl/InfluxDBResultMapper.java
* /opt/cola/permits/1280350859_1646202214.38/0/influxdb-java-2-20-sources-jar/org/influxdb/annotation/Column.java
* /opt/cola/permits/1280350859_1646202214.38/0/influxdb-java-2-20-sources-jar/org/influxdb/InfluxDBMapperException.java
No license file was found, but licenses were detected in source scan.

<name>The MIT License (MIT)</name>

Found in path(s):
* /opt/cola/permits/1280350859_1646202214.38/0/influxdb-java-2-20-sources-jar/META-INF/maven/org.influxdb/influxdb-java/pom.xml

1.195 kerberos 1.16-2ubuntu0.2
1.195.1 Available under license :

[ NOTE: MIT has only incorporated the mechglue and spnego change, and not the incremental propagation changes. The filenames are different between the Sun and MIT sources. The actual MIT filenames appear in the top-level README file. Original text of Sun's LICENSE file follows. ]

Subject to the license set forth below, Sun Microsystems, Inc. donates the attached files to MIT for the purpose of including these modifications and additions in future versions of the Kerberos system.

Many of the files attached are subject to licenses issued by other entities, including OpenVision, MIT, and FundsXpress. See the individual files, and/or related Readme files, for these licenses.

In addition Sun requires that the license set forth below be incorporated into any future version of the Kerberos system which contains portions of the files attached. The following files must be listed, in the top level Readme file, as being provided subject to such license:

cmd/krb5/iprop/iprop.x
cmd/krb5/iprop/iprop_hdr.h
cmd/krb5/kadmin/server/ipropd_svc.c
cmd/krb5/kproplog/kproplog.c
cmd/krb5/slave/kpropd_rpc.c
lib/gss_mechs/mech_krb5/etc/kdb5_err.c
lib/gss_mechs/mech_spnego/mech/gssapiP_spnego.h
lib/gss_mechs/mech_spnego/mecch/spnego_mech.c
lib/krb5/kadm5/kadm_hostsrv_names.c
lib/krb5/krb5/kdb_convert.c
lib/krb5/krb5/kdb_hdr.h
lib/krb5/krb5/kdb_log.c
lib/krb5/krb5/kdb_log.h
lib/libgss/g_accept_sec_context.c
lib/libgss/g_acquire_cred.c
lib/libgss/g_canon_name.c
lib/libgss/g_compare_name.c
lib/libgss/g_context_time.c
lib/libgss/g_delete_sec_context.c
lib/libgss/g_dup_name.c
lib/libgss/g_dup_status.c
lib/libgss/g_imp_name.c
lib/libgss/g_imp_sec_context.c
lib/libgss/g_init_sec_context.c
lib/libgss/g_initialize.c
lib/libgss/g_inquire_context.c
lib/libgss/g_inquire_cred.c
lib/libgss/g_inquire_names.c
lib/libgss/g_process_context.c
lib/libgss/g_rel_buffer.c
lib/libgss/g_rel_cred.c
lib/libgss/g_rel_name.c
lib/libgss/g_rel_oid_set.c
lib/libgss/g_seal.c
lib/libgss/g_sign.c
lib/libgss/g_store_cred.c
lib/libgss/g_unseal.c
lib/libgss/g_userok.c
lib/libgss/g_utils.c
lib/libgss/g_verify.c
lib/libgss/gssd_pname_to_uid.c
uts/common/gssapi/include/gssapi_err_generic.h
uts/common/gssapi/include/mechglueP.h

Sun's License is as follows:

Copyright (c) 2004 Sun Microsystems, Inc.

Permission is hereby granted, free of charge, to any person obtaining a
 copy of this software and associated documentation files (the
 "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, 
distribute, sublicense, and/or sell copies of the Software, and to 
permit persons to whom the Software is furnished to do so, subject to 
the following conditions:

The above copyright notice and this permission notice shall be included 
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS 
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF 
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. 
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY 
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, 
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE 
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright
==========

Copyright © 1985-2020 by the Massachusetts Institute of 
Technology and its contributors. All rights reserved.

See :ref:`mitK5license` for additional copyright and license 
information.

Copyright (C) 1985-2020 by the Massachusetts Institute of Technology.

All rights reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions are 
met:

* Redistributions of source code must retain the above copyright 
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright 
  notice, this list of conditions and the following disclaimer in the 
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS 
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT 
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR 
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT 
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, 
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT 
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, 
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY 
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT 
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Downloading of this software may constitute an export of cryptographic software from the United States of America that is subject to the United States Export Administration Regulations (EAR), 15 CFR 730-774. Additional laws or regulations may apply. It is the responsibility of the person or entity contemplating export to comply with all applicable export laws and regulations, including obtaining any required license from the U.S. government.

The U.S. government prohibits export of encryption source code to certain countries and individuals, including, but not limited to, the countries of Cuba, Iran, North Korea, Sudan, Syria, and residents and nationals of those countries.

Documentation components of this software distribution are licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License. (https://creativecommons.org/licenses/by-sa/3.0/)

Individual source code files are copyright MIT, Cygnus Support, Novell, OpenVision Technologies, Oracle, Red Hat, Sun Microsystems, FundsXpress, and others.

Project Athena, Athena, Athena MUSE, Discuss, Hesiod, Kerberos, Moira, and Zephyr are trademarks of the Massachusetts Institute of Technology (MIT). No commercial use of these trademarks may be made without prior written permission of MIT.

"Commercial use" means use of a name in a product or other for-profit manner. It does NOT prevent a commercial firm from referring to the MIT trademarks in order to convey information (although in doing so, recognition of their trademark status should be given).

======================================================================

The following copyright and permission notice applies to the OpenVision Kerberos Administration system located in "kadmin/create", "kadmin/dbutil", "kadmin/passwd", "kadmin/server", "lib/kadm5", and portions of "lib/rpc":


WARNING: Retrieving the OpenVision Kerberos Administration system source code, as described below, indicates your acceptance of the following terms. If you do not agree to the following terms, do not retrieve the OpenVision Kerberos administration system.
You may freely use and distribute the Source Code and Object Code compiled from it, with or without modification, but this Source Code is provided to you "AS IS" EXCLUSIVE OF ANY WARRANTY, INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY OTHER WARRANTY, WHETHER EXPRESS OR IMPLIED. IN NO EVENT WILL OPENVISION HAVE ANY LIABILITY FOR ANY LOST PROFITS, LOSS OF DATA OR COSTS OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, INCLUDING, WITHOUT LIMITATION, THOSE RESULTING FROM THE USE OF THE SOURCE CODE, OR THE FAILURE OF THE SOURCE CODE TO PERFORM, OR FOR ANY OTHER REASON.

OpenVision retains all copyrights in the donated Source Code. OpenVision also retains copyright to derivative works of the Source Code, whether created by OpenVision or by a third party. The OpenVision copyright notice must be preserved if derivative works are made based on the donated Source Code.

OpenVision Technologies, Inc. has donated this Kerberos Administration system to MIT for inclusion in the standard Kerberos 5 distribution. This donation underscores our commitment to continuing Kerberos technology development and our gratitude for the valuable work which has been performed by MIT and the Kerberos community.

======================================================================
Portions contributed by Matt Crawford "crawdad@fnal.gov" were work performed at Fermi National Accelerator Laboratory, which is operated by Universities Research Association, Inc., under contract DE-AC02-76CHO3000 with the U.S. Department of Energy.

======================================================================
Portions of "src/lib/crypto" have the following copyright:

Copyright (C) 1998 by the FundsXpress, INC.

All rights reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and
without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FundsXpress not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FundsXpress makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

======================================================================

The implementation of the AES encryption algorithm in "src/lib/crypto/builtin/aes" has the following copyright:

    Copyright (C) 2001, Dr Brian Gladman "brg@gladman.uk.net",
    Worcester, UK.
    All rights reserved.

LICENSE TERMS

The free distribution and use of this software in both source and binary form is allowed (with or without changes) provided that:

1. distributions of this source code include the above copyright notice, this list of conditions and the following disclaimer;

2. distributions in binary form include the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other associated materials;

3. the copyright holder's name is not used to endorse products built using this software without specific written permission.

DISCLAIMER

This software is provided 'as is' with no explicit or implied warranties in respect of any properties, including, but not limited to, correctness and fitness for purpose.

======================================================================

Portions contributed by Red Hat, including the pre-authentication plug-in framework and the NSS crypto implementation, contain the following copyright:
Copyright (C) 2006 Red Hat, Inc.
Portions copyright (C) 2006 Massachusetts Institute of Technology
All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.

* Neither the name of Red Hat, Inc., nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRict LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The bundled verto source code is subject to the following license:

Copyright 2011 Red Hat, Inc.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

The MS-KKDCP client implementation has the following copyright:

Copyright 2013,2014 Red Hat, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The implementations of GSSAPI mechglue in GSSAPI-SPNEGO in "src/lib/gssapi", including the following files:
and the initial implementation of incremental propagation, including the following new or changed files:

include/iprop_hdr.h
kadmin/server/ipropd_svc.c
lib/kdb/iprop.x
lib/kdb/kdb_convert.c
lib/kdb/kdb_log.c
lib/kdb/kdb_log.h
lib/krb5/error_tables/kdb5_err.et
kprop/kpropd_rpc.c
kprop/kproplog.c

are subject to the following license:

Copyright (C) 2004 Sun Microsystems, Inc.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

======================================================================

Kerberos V5 includes documentation and software developed at the
University of California at Berkeley, which includes this copyright
notice:

Copyright (C) 1983 Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above
copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Portions contributed by Novell, Inc., including the LDAP database backend, are subject to the following license:

   Copyright (C) 2004-2005, Novell, Inc.
   All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The copyright holder's name is not used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================
Portions funded by Sandia National Laboratory and developed by the University of Michigan's Center for Information Technology Integration, including the PKINIT implementation, are subject to the following license:

COPYRIGHT (C) 2006-2007 THE REGENTS OF THE UNIVERSITY OF MICHIGAN ALL RIGHTS RESERVED

Permission is granted to use, copy, create derivative works and redistribute this software and such derivative works for any purpose, so long as the name of The University of Michigan is not used in any advertising or publicity pertaining to the use of distribution of this software without specific, written prior authorization. If the above copyright notice or any other identification of the University of Michigan is included in any copy of any portion of this software, then the disclaimer below must also be included.

THIS SOFTWARE IS PROVIDED AS IS, WITHOUT REPRESENTATION FROM THE UNIVERSITY OF MICHIGAN AS TO ITS FITNESS FOR ANY PURPOSE, AND WITHOUT WARRANTY BY THE UNIVERSITY OF MICHIGAN OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE REGENTS OF THE UNIVERSITY OF MICHIGAN SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WITH RESPECT TO ANY CLAIM ARISING OUT OF OR IN CONNECTION WITH THE USE OF THE SOFTWARE, EVEN IF IT HAS BEEN OR IS HEREAFTER ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

======================================================================

The pkcs11.h file included in the PKINIT code has the following license:

Copyright 2006 g10 Code GmbH
Copyright 2006 Andreas Jellinghaus
This file is free software; as a special exception the author gives unlimited permission to copy and/or distribute it, with or without modifications, as long as this notice is preserved.

This file is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, to the extent permitted by law; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

======================================================================

Portions contributed by Apple Inc. are subject to the following license:

Copyright 2004-2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

======================================================================

The implementations of UTF-8 string handling in src/util/support and src/lib/krb5/unicode are subject to the following copyright and permission notice:

The OpenLDAP Public License
Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:
1. Redistributions in source form must retain copyright statements and notices,

2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and

3. Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distribute verbatim copies of this document is granted.

======================================================================

Marked test programs in src/lib/krb5/krb have the following copyright:
Copyright (C) 2006 Kungliga Tekniska Högskola 
(Royal Institute of Technology, Stockholm, Sweden). 
All rights reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions 
are met:

1. Redistributions of source code must retain the above 
copyright notice, this list of conditions and the following 
disclaimer.

2. Redistributions in binary form must reproduce the above 
copyright notice, this list of conditions and the following 
disclaimer in the documentation and/or other materials provided 
with the distribution.

3. Neither the name of KTH nor the names of its contributors may 
be used to endorse or promote products derived from this 
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY KTH AND ITS CONTRIBUTORS "AS IS" AND ANY 
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, 
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A 
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL KTH OR ITS 
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, 
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT 
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF 
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON 
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, 
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT 
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The KCM Mach RPC definition file used on macOS has the following 
copyright:

Copyright (C) 2009 Kungliga Tekniska Högskola 
(Royal Institute of Technology, Stockholm, Sweden). 
All rights reserved.

Portions Copyright (C) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without 
modification, are permitted provided that the following conditions 
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Portions of the RPC implementation in src/lib/rpc and src/include/gssrpc have the following copyright and permission notice:

Copyright (C) 2010, Oracle America, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the "Oracle America, Inc." nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Copyright (C) 2006,2007,2009 NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2000 by Carnegie Mellon University

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Carnegie Mellon University not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (C) 2002 Naval Research Laboratory (NRL/CCS)

Permission to use, copy, modify and distribute this software and its documentation is hereby granted, provided that both the copyright notice and this permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof.

NRL ALLOWS FREE USE OF THIS SOFTWARE IN ITS "AS IS" CONDITION AND DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.


Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Cygnus Support makes no
representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

======================================================================

Copyright (C) 2006 Secure Endpoints Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

Portions of the implementation of the Fortuna-like PRNG are subject to the following notice:

Copyright (C) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Copyright (C) 1994 by the University of Southern California

EXPORT OF THIS SOFTWARE from the United States of America may
require a specific license from the United States Government. It
is the responsibility of any person or organization
contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to copy, modify, and distribute
this software and its documentation in source and binary forms is
hereby granted, provided that any documentation or other materials
related to such distribution or use acknowledge that the software
was developed by the University of Southern California.

DISCLAIMER OF WARRANTY. THIS SOFTWARE IS PROVIDED "AS IS". The
University of Southern California MAKES NO REPRESENTATIONS OR
WARRANTIES, EXPRESS OR IMPLIED. By way of example, but not
limitation, the University of Southern California MAKES NO
REPRESENTATIONS OR WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY
PARTICULAR PURPOSE. The University of Southern California shall not
be held liable for any liability nor for any direct, indirect, or
consequential damages with respect to any claim by the user or
distributor of the ksu software.

======================================================================

Copyright (C) 1995
The President and Fellows of Harvard University

This code is derived from software contributed to Harvard by Jeremy
Rassen.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================
Copyright (C) 2008 by the Massachusetts Institute of Technology.  
Copyright 1995 by Richard P. Basch.  All Rights Reserved.  
Copyright 1995 by Lehman Brothers, Inc.  All Rights Reserved.  

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and
this permission notice appear in supporting documentation, and that
the name of Richard P. Basch, Lehman Brothers and M.I.T. not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission. Richard P.
Basch, Lehman Brothers and M.I.T. make no representations about the
suitability of this software for any purpose. It is provided "as
is" without express or implied warranty.

======================================================================

The following notice applies to "src/lib/krb5/krb/strptime.c" and
"src/include/k5-queue.h".

Copyright (C) 1997, 1998 The NetBSD Foundation, Inc.
All rights reserved.

This code was contributed to The NetBSD Foundation by Klaus Klein.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above
copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.

3. All advertising materials mentioning features or use of this
software must display the following acknowledgement:

   This product includes software developed by the NetBSD
   Foundation, Inc. and its contributors.

4. Neither the name of The NetBSD Foundation nor the names of
its contributors may be used to endorse or promote products
derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The following notice applies to Unicode library files in "src/lib/krb5/unicode":

Copyright 1997, 1998, 1999 Computing Research Labs, New Mexico State University

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

The following notice applies to "src/util/support/strlcpy.c":

Copyright (C) 1998 Todd C. Miller "Todd.Miller@courtesan.com"

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL
WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE
AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
NEGligence or OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

======================================================================
The following notice applies to "src/util/profile/argv_parse.c" and
"src/util/profile/argv_parse.h":

Copyright 1999 by Theodore Ts'o.

Permission to use, copy, modify, and distribute this software for
any purpose with or without fee is hereby granted, provided that
the above copyright notice and this permission notice appear in all
copies. THE SOFTWARE IS PROVIDED "AS IS" AND THEODORE TS'O (THE
AUTHOR) DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN
NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT,
INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR
IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. (Isn't
it sick that the U.S. culture of lawsuit-happy lawyers requires
this kind of disclaimer?)

======================================================================
The following notice applies to SWIG-generated code in
"src/util/profile/profile_tcl.c":

Copyright (C) 1999-2000, The University of Chicago

This file may be freely redistributed without license or fee
provided this copyright message remains intact.

======================================================================
The following notice applies to portsion of "src/lib/rpc" and
"src/include/gssrpc":

Copyright (C) 2000 The Regents of the University of Michigan. All
rights reserved.

Copyright (C) 2000 Dug Song "dugsong@UMICH.EDU". All rights
reserved, all wrongs reversed.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Implementations of the MD4 algorithm are subject to the following notice:

Copyright (C) 1990, RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD4 Message Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD4 Message Digest Algorithm" in all material mentioning or referencing the derived work.
RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.

======================================================================
Implementations of the MD5 algorithm are subject to the following
notice:

Copyright (C) 1990, RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it
is identified as the "RSA Data Security, Inc. MD5 Message-Digest
Algorithm" in all material mentioning or referencing this software
or this function.

License is also granted to make and use derivative works provided
that such works are identified as "derived from the RSA Data
Security, Inc. MD5 Message-Digest Algorithm" in all material
mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.

======================================================================

The following notice applies to
"src/lib/crypto/crypto_tests/t_mddriver.c":

Copyright (C) 1990-2, RSA Data Security, Inc. Created 1990. All
rights reserved.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is" without
express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
Portions of "src/lib/krb5" are subject to the following notice:

Copyright (C) 1994 CyberSAFE Corporation.
All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Furthermore if you modify this software you must label your software as modified software and not distribute it in such a fashion that it might be confused with the original M.I.T. software. Neither M.I.T., the Open Computing Security Group, nor CyberSAFE Corporation make any representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Portions contributed by PADL Software are subject to the following license:

Copyright (c) 2011, PADL Software Pty Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.

3. Neither the name of PADL Software nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

======================================================================
The bundled libev source code is subject to the following license:

All files in libev are Copyright (C)2007,2008,2009 Marc Alexander
Lehmann.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Alternatively, the contents of this package may be used under the terms of the GNU General Public License ("GPL") version 2 or any later version, in which case the provisions of the GPL are applicable instead of the above. If you wish to allow the use of your version of this package only under the terms of the GPL and not to allow others to use your version of this file under the BSD license, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL in this and the other files of this package. If you do not delete the provisions above, a recipient may use your version of this file under either the BSD or the GPL.

======================================================================

Files copied from the Intel AESNI Sample Library are subject to the following license:

Copyright (C) 2010, Intel Corporation All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The following notice applies to
"src/ccapi/common/win/OldCC/autolock.hxx":

Copyright (C) 1998 by Danilo Almeida. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright
 notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
 copyright notice, this list of conditions and the following
 disclaimer in the documentation and/or other materials provided
 with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The following notice applies to portions of
"src/plugins/preauth/spake/edwards25519.c" and
"src/plugins/preauth/spake/edwards25519_tables.h":

The MIT License (MIT)

Copyright (c) 2015-2016 the fiat-crypto authors (see the AUTHORS
file).
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

The following notice applies to portions of "src/plugins/preauth/spake/edwards25519.c":

Copyright (c) 2015-2016, Google Inc.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

.. _mitK5license:

MIT Kerberos License information

================================

.. toctree::

:hidden:
1.196 libgcrypt 1.8.1 4ubuntu1.3

1.196.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations
below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.
To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

^L

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that,
in the event an application does not supply such function or
table, the facility still operates, and performs whatever part of
its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable “work that
uses the Library”, as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least
three years, to give the same user the materials specified in
Subsection 6a, above, for a charge no more than the cost of
performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work
      based on the Library, uncombined with any other library
      facilities. This must be distributed under the terms of the
      Sections above.

   b) Give prominent notice with the combined library of the fact
      that part of it is a work based on the Library, and explaining
      where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms
of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Additional license notices for Libgcrypt. -*- org -*-

This file contains the copying permission notices for various files in the Libgcrypt distribution which are not covered by the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL).

These notices all require that a copy of the notice be included in the accompanying documentation and be distributed with binary
distributions of the code, so be sure to include this file along with any binary distributions derived from the GNU C Library.

* BSD_3Clause

For files:
- cipher/sha256-avx-amd64.S
- cipher/sha256-avx2-bmi2-amd64.S
- cipher/sha256-ssse3-amd64.S
- cipher/sha512-avx-amd64.S
- cipher/sha512-avx2-bmi2-amd64.S
- cipher/sha512-ssse3-amd64.S

#+begin_quote
Copyright (c) 2012, Intel Corporation

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY INTEL CORPORATION "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL INTEL CORPORATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#+end_quote
For files:
- random/jitterentropy-base.c
- random/jitterentropy.h
- random/rndjent.c (plus common Libgcrypt copyright holders)

#+begin_quote
* Copyright Stephan Mueller <smueller@chronox.de>, 2013
*
* License
* ========
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, and the entire permission notice in its entirety,
*    including the disclaimer of warranties.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote
*    products derived from this software without specific prior
*    written permission.
*
* ALTERNATIVELY, this product may be distributed under the terms of
* the GNU General Public License, in which case the provisions of the GPL are
* required INSTEAD OF the above restrictions. (This clause is
* necessary due to a potential bad interaction between the GPL and
* the restrictions contained in a BSD-style copyright.)
*
* THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF
* WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT
* OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
* BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
* USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH
* DAMAGE.
#+end_quote

* X License

For files:
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

* Public domain

For files:
- cipher/arcfour-amd64.S

* OCB license 1

For files:
- cipher/cipher-ocb.c

OCB is covered by several patents but may be used freely by most software. See http://web.cs.ucdavis.edu/~rogaway/ocb/license.htm. In particular license 1 is suitable for Libgcrypt: See
License 1  License for Open-Source Software Implementations of OCB  
(Jan 9, 2013)

Under this license, you are authorized to make, use, and distribute open-source software implementations of OCB. This license terminates for you if you sue someone over their open-source software implementation of OCB claiming that you have a patent covering their implementation.

License for Open Source Software Implementations of OCB  
January 9, 2013

1 Definitions

1.1 Licensor means Phillip Rogaway.


1.3 Use means any practice of any invention claimed in the Licensed Patents.

1.4 Software Implementation means any practice of any invention claimed in the Licensed Patents that takes the form of software executing on a user-programmable, general-purpose computer or that takes the form of a computer-readable medium storing such software. Software Implementation does not include, for example, application-specific integrated circuits (ASICs), field-programmable gate arrays (FPGAs), embedded systems, or IP cores.

1.5 Open Source Software means software whose source code is published and made available for inspection and use by anyone because either (a) the source code is subject to a license that permits recipients to copy, modify, and distribute the source code without payment of fees or royalties, or (b) the source code is in the public domain, including code released for public use through a CC0 waiver. All licenses certified by the Open Source Initiative at opensource.org as of January 9, 2013 and all Creative Commons licenses identified on the creativecommons.org website as of January 9,
1.6 Open Source Software Implementation means a Software Implementation in which the software implicating the Licensed Patents is Open Source Software. Open Source Software Implementation does not include any Software Implementation in which the software implicating the Licensed Patents is combined, so as to form a larger program, with software that is not Open Source Software.

2 License Grant

2.1 License. Subject to your compliance with the terms of this license, including the restriction set forth in Section 2.2, Licensor hereby grants to you a perpetual, worldwide, non-exclusive, non-transferable, non-sublicenseable, no-charge, royalty-free, irrevocable license to practice any invention claimed in the Licensed Patents in any Open Source Software Implementation.

2.2 Restriction. If you or your affiliates institute patent litigation (including, but not limited to, a cross-claim or counterclaim in a lawsuit) against any entity alleging that any Use authorized by this license infringes another patent, then any rights granted to you under this license automatically terminate as of the date such litigation is filed.

3 Disclaimer

YOUR USE OF THE LICENSED PATENTS IS AT YOUR OWN RISK AND UNLESS REQUIRED BY APPLICABLE LAW, LICENSOR MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED PATENTS OR ANY PRODUCT EMBODYING ANY LICENSED PATENT, EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL LICENSOR BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM OR RELATED TO ANY USE OF THE LICENSED PATENTS, INCLUDING, WITHOUT LIMITATION, DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR SPECIAL DAMAGES, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES PRIOR TO SUCH AN OCCURRENCE.

1.197 openjsse 1.1.5

1.197.1 Available under license:
The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all
the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation
of the executable. However, as a special exception, the source code
distributed need not include anything that is normally distributed (in either
source or binary form) with the major components (compiler, kernel, and so on)
of the operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source
code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as
expressly provided under this License. Any attempt otherwise to copy, modify,
sublicense or distribute the Program is void, and will automatically terminate
your rights under this License. However, parties who have received copies, or
rights, from you under this License will not have their licenses terminated so
long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it.
However, nothing else grants you permission to modify or distribute the Program
or its derivative works. These actions are prohibited by law if you do not
accept this License. Therefore, by modifying or distributing the Program (or
any work based on the Program), you indicate your acceptance of this License to
do so, and all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program),
the recipient automatically receives a license from the original licensor to
copy, distribute or modify the Program subject to these terms and conditions.
You may not impose any further restrictions on the recipients' exercise of the
rights granted herein. You are not responsible for enforcing compliance by
third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues), conditions
are imposed on you (whether by court order, agreement or otherwise) that
contradict the conditions of this License, they do not excuse you from the
conditions of this License. If you cannot distribute so as to satisfy
simultaneously your obligations under this License and any other pertinent
obligations, then as a consequence you may not distribute the Program at all.
For example, if a patent license would not permit royalty-free redistribution
of the Program by all those who receive copies directly or indirectly through
you, then the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE
STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules,
and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.198 cyrus-sasl 2.1.27~101-g0780600+dfsg-3ubuntu2.4

1.198.1 Available under license:

APPLE PUBLIC SOURCE LICENSE
Version 1.1 - April 19, 1999

Please read this License carefully before downloading this software. By downloading and using this software, you are agreeing to be bound by the terms of this License. If you do not or cannot agree to the terms of this License, please do not download or use the software.

1. General; Definitions. This License applies to any program or other work which Apple Computer, Inc. ("Apple") publicly announces as subject to this Apple Public Source License and which contains a notice placed by Apple identifying such program or work as "Original Code" and stating that it is subject to the terms of this Apple Public Source License version 1.1 (or subsequent version thereof), as it may be revised from time to time by Apple ("License"). As used in this License:

1.1 "Affected Original Code" means only those specific portions of Original Code that allegedly infringe upon any party's intellectual property rights or are otherwise the subject of a claim of infringement.

1.2 "Applicable Patent Rights" mean: (a) in the case where Apple is the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to Apple and (ii) that cover subject matter contained in the Original Code, but only to the extent necessary to use, reproduce and/or distribute the Original Code without infringement; and (b) in the case where You are the grantor of rights, (i) claims of patents that are now or hereafter acquired, owned by or assigned to You and (ii) that cover subject matter in Your Modifications, taken alone or in combination with Original Code.

1.3 "Covered Code" means the Original Code, Modifications, the combination of Original Code and any Modifications, and/or any
respective portions thereof.

1.4 "Deploy" means to use, sublicense or distribute Covered Code other than for Your internal research and development (R&D), and includes without limitation, any and all internal use or distribution of Covered Code within Your business or organization except for R&D use, as well as direct or indirect sublicensing or distribution of Covered Code by You to any third party in any form or manner.

1.5 "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.6 "Modifications" mean any addition to, deletion from, and/or change to, the substance and/or structure of Covered Code. When code is released as a series of files, a Modification is: (a) any addition to or deletion from the contents of a file containing Covered Code; and/or (b) any new file or other representation of computer program statements that contains any part of Covered Code.

1.7 "Original Code" means (a) the Source Code of a program or other work as originally made available by Apple under this License, including the Source Code of any updates or upgrades to such programs or works made available by Apple under this License, and that has been expressly identified by Apple as such in the header file(s) of such work; and (b) the object code compiled from such Source Code and originally made available by Apple under this License.

1.8 "Source Code" means the human readable form of a program or other work that is suitable for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an executable (object code).

1.9 "You" or "Your" means an individual or a legal entity exercising rights under this License. For legal entities, "You" or "Your" includes any entity which controls, is controlled by, or is under common control with, You, where "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of fifty percent (50%) or more of the outstanding shares or beneficial ownership of such entity.

2. Permitted Uses; Conditions & Restrictions. Subject to the terms and conditions of this License, Apple hereby grants You, effective on the date You accept this License and download the Original Code, a world-wide, royalty-free, non- exclusive license, to the extent of Apple's Applicable Patent Rights and copyrights covering the Original Code, to do the following:
2.1 You may use, copy, modify and distribute Original Code, with or without Modifications, solely for Your internal research and development, provided that You must in each instance:

(a) retain and reproduce in all copies of Original Code the copyright and other proprietary notices and disclaimers of Apple as they appear in the Original Code, and keep intact all notices in the Original Code that refer to this License;

(b) include a copy of this License with every copy of Source Code of Covered Code and documentation You distribute, and You may not offer or impose any terms on such Source Code that alter or restrict this License or the recipients' rights hereunder, except as permitted under Section 6; and

(c) completely and accurately document all Modifications that you have made and the date of each such Modification, designate the version of the Original Code you used, prominently include a file carrying such information with the Modifications, and duplicate the notice in Exhibit A in each file of the Source Code of all such Modifications.

2.2 You may Deploy Covered Code, provided that You must in each instance:

(a) satisfy all the conditions of Section 2.1 with respect to the Source Code of the Covered Code;

(b) make all Your Deployed Modifications publicly available in Source Code form via electronic distribution (e.g. download from a web site) under the terms of this License and subject to the license grants set forth in Section 3 below, and any additional terms You may choose to offer under Section 6. You must continue to make the Source Code of Your Deployed Modifications available for as long as you Deploy the Covered Code or twelve (12) months from the date of initial Deployment, whichever is longer;

(c) if You Deploy Covered Code containing Modifications made by You, inform others of how to obtain those Modifications by filling out and submitting the information found at http://www.apple.com/publicsource/modifications.html, if available; and

(d) if You Deploy Covered Code in object code, executable form only, include a prominent notice, in the code itself as well as in related documentation, stating that Source Code of the Covered Code is available under the terms of this License with information on how and where to obtain such Source Code.
3. Your Grants. In consideration of, and as a condition to, the licenses granted to You under this License:

(a) You hereby grant to Apple and all third parties a non-exclusive, royalty-free license, under Your Applicable Patent Rights and other intellectual property rights owned or controlled by You, to use, reproduce, modify, distribute and Deploy Your Modifications of the same scope and extent as Apple's licenses under Sections 2.1 and 2.2; and

(b) You hereby grant to Apple and its subsidiaries a non-exclusive, worldwide, royalty-free, perpetual and irrevocable license, under Your Applicable Patent Rights and other intellectual property rights owned or controlled by You, to use, reproduce, execute, compile, display, perform, modify or have modified (for Apple and/or its subsidiaries), sublicense and distribute Your Modifications, in any form, through multiple tiers of distribution.

4. Larger Works. You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In each such instance, You must make sure the requirements of this License are fulfilled for the Covered Code or any portion thereof.

5. Limitations on Patent License. Except as expressly stated in Section 2, no other patent rights, express or implied, are granted by Apple herein. Modifications and/or Larger Works may require additional patent licenses from Apple which Apple may grant in its sole discretion.

6. Additional Terms. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations and/or other rights consistent with the scope of the license granted herein ("Additional Terms") to one or more recipients of Covered Code. However, You may do so only on Your own behalf and as Your sole responsibility, and not on behalf of Apple. You must obtain the recipient's agreement that any such Additional Terms are offered by You alone, and You hereby agree to indemnify, defend and hold Apple harmless for any liability incurred by or claims asserted against Apple by reason of any such Additional Terms.

7. Versions of the License. Apple may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Once Original Code has been published under a particular version of this License, You may continue to use it under the terms of that version. You may also choose to use such Original Code under the terms of any subsequent version of this
License published by Apple. No one other than Apple has the right to
modify the terms applicable to Covered Code created under this
License.

8. NO WARRANTY OR SUPPORT. The Original Code may contain in whole or
in part pre-release, untested, or not fully tested works. The
Original Code may contain errors that could cause failures or loss of
data, and may be incomplete or contain inaccuracies. You expressly
acknowledge and agree that use of the Original Code, or any portion
thereof, is at Your sole and entire risk. THE ORIGINAL CODE IS
PROVIDED "AS IS" AND WITHOUT WARRANTY, UPGRADES OR SUPPORT OF ANY KIND
AND APPLE AND APPLE'S LICENSOR(S) (FOR THE PURPOSES OF SECTIONS 8 AND
9, APPLE AND APPLE'S LICENSOR(S) ARE COLLECTIVELY REFERRED TO AS
"APPLE") EXPRESSLY DISCLAIM ALL WARRANTIES AND/OR CONDITIONS, EXPRESS
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
AND/OR CONDITIONS OF MERCHANTABILITY OR SATISFACTORY QUALITY AND
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY
RIGHTS. APPLE DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE
ORIGINAL CODE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF
THE ORIGINAL CODE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT
DEFECTS IN THE ORIGINAL CODE WILL BE CORRECTED. NO ORAL OR WRITTEN
INFORMATION OR ADVICE GIVEN BY APPLE OR AN APPLE AUTHORIZED
REPRESENTATIVE SHALL CREATE A WARRANTY OR IN ANY WAY INCREASE THE
SCOPE OF THIS WARRANTY. You acknowledge that the Original Code is not
intended for use in the operation of nuclear facilities, aircraft
navigation, communication systems, or air traffic control machines in
which case the failure of the Original Code could lead to death,
personal injury, or severe physical or environmental damage.

9. Liability.

9.1 Infringement. If any portion of, or functionality implemented by,
the Original Code becomes the subject of a claim of infringement,
Apple may, at its option: (a) attempt to procure the rights necessary
for Apple and You to continue using the Affected Original Code; (b)
modify the Affected Original Code so that it is no longer infringing;
or (c) suspend Your rights to use, reproduce, modify, sublicense and
distribute the Affected Original Code until a final determination of
the claim is made by a court or governmental administrative agency of
competent jurisdiction and Apple lifts the suspension as set forth
below. Such suspension of rights will be effective immediately upon
Apple's posting of a notice to such effect on the Apple web site that
is used for implementation of this License. Upon such final
determination being made, if Apple is legally able, without the
payment of a fee or royalty, to resume use, reproduction,
modification, sublicensing and distribution of the Affected Original
Code, Apple will lift the suspension of rights to the Affected
Original Code by posting a notice to such effect on the Apple web site
that is used for implementation of this License. If Apple suspends Your rights to Affected Original Code, nothing in this License shall be construed to restrict You, at Your option and subject to applicable law, from replacing the Affected Original Code with non-infringing code or independently negotiating for necessary rights from such third party.

9.2 LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES SHALL APPLE BE LIABLE FOR ANY INCIDENTAL, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES ARISING OUT OF OR RELATING TO THIS LICENSE OR YOUR USE OR INABILITY TO USE THE ORIGINAL CODE, OR ANY PORTION THEREOF, WHETHER UNDER A THEORY OF CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCTS LIABILITY OR OTHERWISE, EVEN IF APPLE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF ESSENTIAL PURPOSE OF ANY REMEDY. In no event shall Apple's total liability to You for all damages under this License exceed the amount of fifty dollars ($50.00).

10. Trademarks. This License does not grant any rights to use the trademarks or trade names "Apple", "Apple Computer", "Mac OS X", "Mac OS X Server" or any other trademarks or trade names belonging to Apple (collectively "Apple Marks") and no Apple Marks may be used to endorse or promote products derived from the Original Code other than as permitted by and in strict compliance at all times with Apple's third party trademark usage guidelines which are posted at http://www.apple.com/legal/guidelinesfor3rdparties.html.

11. Ownership. Apple retains all rights, title and interest in and to the Original Code and any Modifications made by or on behalf of Apple ("Apple Modifications"), and such Apple Modifications will not be automatically subject to this License. Apple may, at its sole discretion, choose to license such Apple Modifications under this License, or on different terms from those contained in this License or may choose not to license them at all. Apple's development, use, reproduction, modification, sublicensing and distribution of Covered Code will not be subject to this License.

12. Termination.

12.1 Termination. This License and the rights granted hereunder will terminate:

(a) automatically without notice from Apple if You fail to comply with any term(s) of this License and fail to cure such breach within 30 days of becoming aware of such breach; (b) immediately in the event of the circumstances described in Section 13.5(b); or (c) automatically without notice from Apple if You, at any time during the term of this License, commence an action for patent infringement against Apple.
12.2 Effect of Termination. Upon termination, You agree to immediately stop any further use, reproduction, modification, sublicensing and distribution of the Covered Code and to destroy all copies of the Covered Code that are in your possession or control. All sublicenses to the Covered Code which have been properly granted prior to termination shall survive any termination of this License. Provisions which, by their nature, should remain in effect beyond the termination of this License shall survive, including but not limited to Sections 3, 5, 8, 9, 10, 11, 12.2 and 13. Neither party will be liable to the other for compensation, indemnity or damages of any sort solely as a result of terminating this License in accordance with its terms, and termination of this License will be without prejudice to any other right or remedy of either party.


13.1 Government End Users. The Covered Code is a "commercial item" as defined in FAR 2.101. Government software and technical data rights in the Covered Code include only those rights customarily provided to the public as defined in this License. This customary commercial license in technical data and software is provided in accordance with FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for Department of Defense purchases, DFAR 252.227-7015 (Technical Data -- Commercial Items) and 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation). Accordingly, all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

13.2 Relationship of Parties. This License will not be construed as creating an agency, partnership, joint venture or any other form of legal association between You and Apple, and You will not represent to the contrary, whether expressly, by implication, appearance or otherwise.

13.3 Independent Development. Nothing in this License will impair Apple's right to acquire, license, develop, have others develop for it, market and/or distribute technology or products that perform the same or similar functions as, or otherwise compete with, Modifications, Larger Works, technology or products that You may develop, produce, market or distribute.

13.4 Waiver; Construction. Failure by Apple to enforce any provision of this License will not be deemed a waiver of future enforcement of that or any other provision. Any law or regulation which provides that the language of a contract shall be construed against the drafter will not apply to this License.
13.5 Severability. (a) If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of the License will be enforced to the maximum extent permissible so as to effect the economic benefits and intent of the parties, and the remainder of this License will continue in full force and effect. (b) Notwithstanding the foregoing, if applicable law prohibits or restricts You from fully and/or specifically complying with Sections 2 and/or 3 or prevents the enforceability of either of those Sections, this License will immediately terminate and You must immediately discontinue any use of the Covered Code and destroy all copies of it that are in your possession or control.

13.6 Dispute Resolution. Any litigation or other dispute resolution between You and Apple relating to this License shall take place in the Northern District of California, and You and Apple hereby consent to the personal jurisdiction of, and venue in, the state and federal courts within that District with respect to this License. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded.

13.7 Entire Agreement; Governing Law. This License constitutes the entire agreement between the parties with respect to the subject matter hereof. This License shall be governed by the laws of the United States and the State of California, except that body of California law concerning conflicts of law.

Where You are located in the province of Quebec, Canada, the following clause applies: The parties hereby confirm that they have requested that this License and all related documents be drafted in English. Les parties ont exige que le present contrat et tous les documents connexes soient rediges en anglais.

EXHIBIT A.

"Portions Copyright (c) 1999 Apple Computer, Inc. All Rights Reserved. This file contains Original Code and/or Modifications of Original Code as defined in and that are subject to the Apple Public Source License Version 1.1 (the "License"). You may not use this file except in compliance with the License. Please obtain a copy of the License at http://www.apple.com/publicsource and read it before using this file.

The Original Code and all software distributed under the License are distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. Please see the
License for the specific language governing rights and limitations
under the License."
/* CMU libsasl
 * Tim Martin
 * Rob Earhart
 * Rob Siemborski
 */
/*
 * Copyright (c) 1998-2003 Carnegie Mellon University. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The name "Carnegie Mellon University" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For permission or any other legal
 *    details, please contact
 *    Office of Technology Transfer
 *    Carnegie Mellon University
 *    5000 Forbes Avenue
 *    Pittsburgh, PA 15213-3890
 *    (412) 268-4387, fax: (412) 268-7395
 *    tech-transfer@andrew.cmu.edu
 *
 * 4. Redistributions of any form whatsoever must retain the following
 *    acknowledgment:
 *    "This product includes software developed by Computing Services
 *    at Carnegie Mellon University (http://www.cmu.edu/computing/)."
 *
 * CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO
 * THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
 * AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE
 * FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
 * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN
 * AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING
 * OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
 */
Copyright (C) 1995-1997 Eric Young (eay@mincom.oz.au)
All rights reserved.
This package is an DES implementation written by Eric Young (eay@mincom.oz.au). The implementation was written so as to conform with MIT's libdes.

This library is free for commercial and non-commercial use as long as the following conditions are heard to. The following conditions apply to all code found in this distribution.

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of that the SSL library. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
   This product includes software developed by Eric Young (eay@mincom.oz.au)

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License.]

The reason behind this being stated in this direct manner is past experience in code simply being copied and the attribution removed from it and then being distributed as part of other packages. This implementation was a non-trivial and unpaid effort.
/* CMU libasl
 * Tim Martin
 * Rob Earhart
 * Rob Siemborski
 */

/*
 * Copyright (c) 2001 Carnegie Mellon University. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The name "Carnegie Mellon University" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For permission or any other legal
 *    details, please contact
 *       Office of Technology Transfer
 *       Carnegie Mellon University
 *       5000 Forbes Avenue
 *       Pittsburgh, PA 15213-3890
 *       (412) 268-4387, fax: (412) 268-7395
 *       tech-transfer@andrew.cmu.edu
 *
 * 4. Redistributions of any form whatsoever must retain the following
 *    acknowledgment:
 *       "This product includes software developed by Computing Services
 *       at Carnegie Mellon University (http://www.cmu.edu/computing/)."
 *
 * CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO
 * THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
 * AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE
 * FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
 * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN
 * AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING
 * OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
 */
1.199 netty-transport-native-kqueue

4.1.46.Final

1.199.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-jar/io/netty/channel/kqueue/KQueueEventArray.java
  */

* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-jar/io/netty/channel/kqueue/KQueueEventArray.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 1761
jar/io/netty/channel/kqueue/AcceptFilter.java
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-
ar/io/netty/channel/kqueue/AbstractKQueueServerChannel.java
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-
ar/io/netty/channel/kqueue/KQueueDatagramChannel.java
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-
ar/io/netty/channel/kqueue/KQueueRecvByteAllocatorHandle.java
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-
ar/io/netty/channel/kqueue/AbstractKQueueStreamChannel.java
* /opt/cola/permits/1283432362_1646722053.88/0/netty-transport-native-kqueue-4-1-46-final-sources-
ar/io/netty/channel/kqueue/KQueueEventLoop.java

1.200 jackson-annotations 2.10.0
1.200.1 Available under license:
This copy of Jackson JSON processor annotations is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.201 httpcomponents-client 4.5.12
1.201.1 Available under license:
Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the 
Work (including but not limited to damages for loss of goodwill, 
work stoppage, computer failure or malfunction, or any and all 
other commercial damages or losses), even if such Contributor 
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing 
the Work or Derivative Works thereof, You may choose to offer, 
and charge a fee for, acceptance of support, warranty, indemnity, 
or other liability obligations and/or rights consistent with this 
License. However, in accepting such obligations, You may act only 
on Your own behalf and on Your sole responsibility, not on behalf 
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

========================================================================= 
This project includes Public Suffix List copied from 
<https://publicsuffix.org/list/effective_tld_names.dat> 
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
means each individual or legal entity that creates, contributes to 
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used 
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached 
the notice in Exhibit A, the Executable Form of such Source Code 
Form, and Modifications of such Source Code Form, in each case
including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions
----------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:
(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this
License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any
liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
* 6. Disclaimer of Warranty
* ____________________________
* 
* Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.
* 
************************************************************************
************************************************************************
* 7. Limitation of Liability
* ____________________________
* 
* Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.
* 
************************************************************************
************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

-------------------------------------------
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.202 nss 3.35-2ubuntu2.13

1.202.1 Available under license :

/* Copyright 2016-2017 INRIA and Microsoft Corporation
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

NSS is available under the Mozilla Public License, version 2, a copy of which is below.

Note on GPL Compatibility

The MPL 2, section 3.3, permits you to combine NSS with code under the GNU General Public License (GPL) version 2, or any later version of that license, to make a Larger Work, and distribute the result under the GPL. The only condition is that you must also make NSS, and any changes you have made to it, available to recipients under the terms of the MPL 2 also.

Anyone who receives the combined code from you does not have to continue
to dual licence in this way, and may, if they wish, distribute under the terms of either of the two licences - either the MPL alone or the GPL alone. However, we discourage people from distributing copies of NSS under the GPL alone, because it means that any improvements they make cannot be reincorporated into the main version of NSS. There is never a need to do this for license compatibility reasons.

Note on LGPL Compatibility
--------------------------

The above also applies to combining MPLed code in a single library with code under the GNU Lesser General Public License (LGPL) version 2.1, or any later version of that license. If the LGPLed code and the MPLed code are not in the same library, then the copyleft coverage of the two licences does not overlap, so no issues arise.

Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
 means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
 means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
 means Covered Software of a particular Contributor.

1.4. "Covered Software"
 means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
 means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the
1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in
a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently, any and
all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to,
deletion from, or modification of the contents of Covered
Software; or

(b) any new file in Source Code Form that contains any Covered
Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method,
process, and apparatus claims, in any patent Licensable by such
Contributor that would be infringed, but for the grant of the
License, by the making, using, selling, offering for sale, having
made, import, or transfer of either its Contributions or its
Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU
Lesser General Public License, Version 2.1, the GNU Affero General
Public License, Version 3.0, or any later versions of those
licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that
controls, is controlled by, or is under common control with You. For
purposes of this definition, “control” means (a) the power, direct
or indirect, to cause the direction or management of such entity,
whether by contract or otherwise, or (b) ownership of more than
fifty percent (50%) of the outstanding shares or beneficial
ownership of such entity.

2. License Grants and Conditions
---------------------------------

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free,
non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)
Licensable by such Contributor to use, reproduce, make available,
modify, display, perform, distribute, and otherwise exploit its
Contributions, either on an unmodified basis, with Modifications, or
as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer
for sale, have made, import, and otherwise transfer either its
Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution
become effective for each Contribution on the date the Contributor first
distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under
this License. No additional rights or licenses will be implied from the
distribution or licensing of Covered Software under this License.
Notwithstanding Section 2.1(b) above, no patent license is granted by a
Contributor:

(a) for any code that a Contributor has removed from Covered Software;
or

(b) for infringements caused by: (i) Your and any other third party’s
modifications of Covered Software, or (ii) the combination of its
Contributions with other software (except as part of its Contributor
Version); or

(c) under Patent Claims infringed by Covered Software in the absence of
its Contributions.
This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

-------------------

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of
the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to
statute, judicial order, or regulation then You must: (a) comply with
the terms of this License to the maximum extent possible; and (b)
describe the limitations and the code they affect. Such description must
be placed in a text file included with all distributions of the Covered
Software under this License. Except to the extent prohibited by statute
or regulation, such description must be sufficiently detailed for a
recipient of ordinary skill to be able to understand it.

5. Termination

-------------

5.1. The rights granted under this License will terminate automatically
if You fail to comply with any of its terms. However, if You become
compliant, then the rights granted under this License from a particular
Contributor are reinstated (a) provisionally, unless and until such
Contributor explicitly and finally terminates Your grants, and (b) on an
ongoing basis, if such Contributor fails to notify You of the
non-compliance by some reasonable means prior to 60 days after You have
come back into compliance. Moreover, Your grants from a particular
Contributor are reinstated on an ongoing basis if such Contributor
notifies You of the non-compliance by some reasonable means, this is the
first time You have received notice of non-compliance with this License
from such Contributor, and You become compliant prior to 30 days after
Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent
infringement claim (excluding declaratory judgment actions,
counter-claims, and cross-claims) alleging that a Contributor Version
directly or indirectly infringes any patent, then the rights granted to
You by any and all Contributors for the Covered Software under Section
2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all
end user license agreements (excluding distributors and resellers) which
have been validly granted by You or Your distributors under this License
prior to termination shall survive termination.

********************************************************************************
*                                                                                          *
*  6. Disclaimer of Warranty                                                                 *
*  ____________________                                                                  *
*                                                                                          *
*  * Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. *
*  ********************************************************************************
7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
of the License under which You originally received the Covered Software,
or under the terms of any subsequent version published by the license
steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a
modified version of this License if you rename the license and remove
any references to the name of the license steward (except to note that
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary
Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.
Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.
This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.
Within this directory, each of the file listed below is licensed under the terms given in the file LICENSE-MPL, also in this directory.

basecvt.pod
gcd.pod
# This file contains a list of people who’ve made non-trivial
# contribution to the Google C++ Testing Framework project. People
# who commit code to the project are encouraged to add their names
# here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@gmail.com>
Ji Sigursson <joi@gmail.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keithray@gmail.com>
Kenton Varda <kenton@gmail.com>
Manuel Klimek <klimek@gmail.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phenna@gmail.com>
Patrick Riley <pfrr@gmail.com>
Peter Kaminski <piotr@gmail.com>
Preston Jackson <preston aka jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@gmail.com>
Russ Rufer <ruess@pentad.com>
Sean Mcafee <eefaccm@gmail.com>
Sigurur sgeirsson <siggi@gmail.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@gmail.com>
Vlad Losev <vladl@gmail.com>
Zhanyong Wan <wan@gmail.com>
1.204 cron-utils 9.1.3

1.204.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications...
represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2014 Joze Martin Rozanec

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.205 grep 3.1 2build1
1.205.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.
To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.
"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section

7. This requirement modifies the requirement in section 4 to "keep intact all notices".
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
c) Convey individual copies of the object code with a copy of the
written offer to provide the Corresponding Source. This
alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord
with subsection 6b.

d) Convey the object code by offering access from a designated
place (gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to
copy the object code is a network server, the Corresponding Source
may be on a different server (operated by you or a third party)
that supports equivalent copying facilities, provided you maintain
clear directions next to the object code saying where to find the
Corresponding Source. Regardless of what server hosts the
Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided
you inform other peers where the object code and Corresponding
Source of the work are being offered to the general public at no
charge under subsection 6d.

A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family,
or household purposes, or (2) anything designed or sold for incorporation
into a dwelling. In determining whether a product is a consumer product,
doubtful cases shall be resolved in favor of coverage. For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product. A product
is a consumer product regardless of whether the product has substantial
commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source. The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.
If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:
a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the
covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

1.206 jaxb-core 2.3.1
1.206.1 Available under license :
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
Version 1.0

* 

1. Definitions.

   o 

1.1. ?Contributor? means each individual or entity that creates or contributes to the creation of Modifications.

   o 

1.2. ?Contributor Version? means the combination of the Original Software, prior Modifications used by a
Contributor (if any), and the Modifications made by that particular Contributor.

   o 

1.3. ?Covered Software? means (a) the Original Software, or (b) Modifications, or (c) the combination of files
containing Original Software with files containing Modifications, in each case including portions thereof.

   o 

1.4. ?Executable? means the Covered Software in any form other than Source Code.
1.5. ?Initial Developer? means the individual or entity that first makes Original Software available under this License.

1.6. ?Larger Work? means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. ?License? means this document.

1.8. ?Licensable? means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. ?Modifications? means the Source Code and Executable form of any of the following:
   +
   A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
   +
   B. Any new file that contains any part of the Original Software or previous Modification; or
   +
   C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. ?Original Software? means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. ?Patent Claims? means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. ?Source Code? means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. ?You? (or ?Your?) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, ?You? includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, ?control? means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

*
2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes
or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient’s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.
5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS? BASIS, WITHOUT
WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A
PARTicular PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND
PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER
CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION.
THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE
OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with
terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their
nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial
Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to
as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a
Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes
any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if
the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall,
upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day
notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software
against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly
granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any
distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING
NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER
CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF
SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES
FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR
MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH
PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a “commercial item,” as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of “commercial computer software” (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and “commercial computer software documentation” as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

1.207 shiro-lang 1.7.1
1.207.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of Your accepting any such warranty or additional liability.
APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.208 gnupg 2.2.4 1ubuntu1.4

1.208.1 Available under license:

LICENSE TERMS

Copyright (c) 2000, Dimitrios Souflis
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Dimitrios Souflis nor the names of the
contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Here is a list with collected copyright notices. For details see the description of each individual package.  [Compiled by wk 2017-11-07]

GNUPG is

Copyright (C) 1997-2017 Werner Koch
Copyright (C) 1994-2017 Free Software Foundation, Inc.
Copyright (C) 2003-2017 g10 Code GmbH
Copyright (C) 2002 Klarlvdalens Datakonsult AB
Copyright (C) 1995-1997, 2000-2007 Ulrich Drepper <drepper@gnu.ai.mit.edu>
Copyright (C) 1994 X Consortium
Copyright (C) 1998 by The Internet Society.
Copyright (C) 1998-2004 The OpenLDAP Foundation
Copyright (C) 1998-2004 Kurt D. Zeilenga.
Copyright (C) 1998-2004 Net Boolean Incorporated.
Copyright (C) 2001-2004 IBM Corporation.
Copyright (C) 1999-2003 Howard Y.H. Chu.
Copyright (C) 1999-2003 Symas Corporation.
Copyright (C) 1998-2003 Hallvard B. Furuseth.
Copyright (C) 1992-1996 Regents of the University of Michigan.
Copyright (C) 2000 Dimitrios Souflis
Copyright (C) 2008,2009,2010,2012-2016 William Ahern

GnuPG is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

GnuPG is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
You should have received a copy of the GNU General Public License along with this program; if not, see <https://www.gnu.org/licenses/>.

LIBGCRYPT is

Copyright (C) 1989,1991-2017 Free Software Foundation, Inc.
Copyright (C) 1994 X Consortium
Copyright (C) 1996 L. Peter Deutsch
Copyright (C) 1997 Werner Koch
Copyright (C) 1998 The Internet Society
Copyright (C) 1996-1999 Peter Gutmann, Paul Kendall, and Chris Wedgwood
Copyright (C) 1996-2006 Peter Gutmann, Matt Thomlinson and Blake Coverett
Copyright (C) 2003 Nikos Mavroyanopoulos
Copyright (C) 2006-2007 NTT (Nippon Telegraph and Telephone Corporation)
Copyright (C) 2012-2017 g10 Code GmbH
Copyright (C) 2012 Simon Josefsson, Niels Miller
Copyright (c) 2012 Intel Corporation
Copyright (C) 2013 Christian Grothoff
Copyright (C) 2013-2017 Jussi Kivilinna
Copyright (C) 2013-2014 Dmitry Eremin-Solenikov
Copyright (C) 2014 Stephan Mueller
Copyright (C) 2017 Bundesamt für Sicherheit in der Informationstechnik

Libgcrypt is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

Libgcrypt is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this program; if not, see <http://www.gnu.org/licenses/>.

LIBGPG-ERROR is


libgpg-error is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

libgpg-error is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License
along with this program; if not, see <http://www.gnu.org/licenses/>.

LIBASSUAN is

Copyright (C) 1992-2013 Free Software Foundation, Inc.
Copyright (C) 1994 X Consortium
Copyright (C) 2000 Werner Koch (dd9jn)
Copyright (C) 2001-2016 g10 Code GmbH
Copyright (C) 2004 Simon Josefsson

Assuan is free software; you can redistribute it and/or modify it
under the terms of the GNU Lesser General Public License as
published by the Free Software Foundation; either version 2.1 of
the License, or (at your option) any later version.

Assuan is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this program; if not, see <http://www.gnu.org/licenses/>.

LIBKSBA is

Copyright (C) 2000, 2001 Fabio Fiorina

The library and the header files are distributed under the following
terms (LGPLv3+/GPLv2+):

KSBA is free software; you can redistribute it and/or modify
it under the terms of either

- the GNU Lesser General Public License as published by the Free
  Software Foundation; either version 3 of the License, or (at
  your option) any later version.

or
- the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

or both in parallel, as here.

KSBA is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

The other parts (e.g. manual, build system, tests) are distributed under the following terms (GPLv3):

KSBA is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

KSBA is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

NPTH is

Copyright (C) 2011, 2012, 2014, 2015, 2017 g10 Code GmbH

nPth is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

nPth is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this program; if not, see <https://www.gnu.org/licenses/>.

NTBTLS is

Copyright (C) 2006-2014 Brainspark B.V.
Copyright (C) 2014-2017 g10 Code GmbH
NTBTLS is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

NTBTLS is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, see <http://www.gnu.org/licenses/>.

PINENTRY is

Copyright (C) 1999 Robert Bihlmeyer <robbe@orcus.priv.at>
Copyright (C) 2002, 2008 Klarlvdalens Datakonsult AB (KDAB)
Copyright (C) 2004 by Albrecht Dre <albrecht.dress@arcor.de>
Copyright 2007 Ingo Klcker
Copyright (C) 2014 Serge Voilokov
Copyright (C) 2015 Daiki Ueno
Copyright (C) 2015 Daniel Kahn Gillmor <dkg@fifthhorseman.net>
Copyright 2016 Intevation GmbH

PINENTRY is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

PINENTRY is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, see <https://www.gnu.org/licenses/>.

GPGME is

Copyright (C) 1991-2013 Free Software Foundation, Inc.
Copyright (C) 2000-2001 Werner Koch
Copyright (C) 2001-2017 g10 Code GmbH
Copyright (C) 2002 Klarlvdalens Datakonsult AB
Copyright (C) 2004-2008 Igor Belyi
Copyright (C) 2002 John Goerzen
Copyright (C) 2014, 2015 Martin Albrecht
Copyright (C) 2015 Ben McGinnes
Copyright (C) 2015-2016 Bundesamt fü r Sicherheit in der Informationstechnik
Copyright (C) 2016 Intevation GmbH

GPGME is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

GPGME is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this program; if not, see <http://www.gnu.org/licenses/>.

NSIS is

Copyright 1999-2009 Nullsoft and Contributors
Copyright 2002-2008 Amir Szekely
Copyright 2003 Ramon

This license applies to everything in the NSIS package, except where otherwise noted.

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.
The user interface used with the installer is

Copyright 2002-2009 Joost Verburg

[It is distributed along with NSIS and the same conditions as stated above apply]

TinySCHEME is part of the GnuPG package and is

Copyright (c) 2000, Dimitrios Souflis
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Dimitrios Souflis nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

LIBDNS is part of the GnuPG package and is

Copyright (c) 2008, 2009, 2010, 2012-2016 William Ahern

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ZLIB is

(C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

BZIP2 is

This program, "bzip2", the associated library "libbzip2", and all documentation, are copyright (C) 1996-2010 Julian R Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

SQLite has been put into the public-domain by its author D. Richard Hipp:
The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:

May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.
[Note that only a few files are distributed under this license.]

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating
Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.
1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.
However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
donate the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply, and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
[Note that only a few files are distributed under this license.]

Creative Commons Legal Code

CC0 1.0 Universal

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS DOCUMENT DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever
and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
ii. moral rights retained by the original author(s) and/or performer(s);
iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;
v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and
vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each
member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.

b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.

c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.

d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying,
distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those
subprograms and other parts of the work.

The Corresponding Source need not include anything that users
can regenerate automatically from other parts of the Corresponding
Source.

The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of
copyright on the Program, and are irrevocable provided the stated
conditions are met. This License explicitly affirms your unlimited
permission to run the unmodified Program. The output from running a
covered work is covered by this License only if the output, given its
content, constitutes a covered work. This License acknowledges your
rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not
convey, without conditions so long as your license otherwise remains
in force. You may convey covered works to others for the sole purpose
of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
not control copyright. Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under
the conditions stated below. Sublicensing is not allowed; section 10
makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article
11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such
measures.

When you convey a covered work, you waive any legal power to forbid
circumvention of technological measures to the extent such circumvention
is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or
modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.
Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of
it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently
reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted
by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying
the work, and under which the third party grants, to any of the
parties who would receive the covered work from you, a discriminatory
patent license (a) in connection with copies of the covered work
conveyed by you (or copies made from those copies), or (b) primarily
for and in connection with specific products or compilations that
contain the covered work, unless you entered into that arrangement,
or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting
any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot convey a
covered work so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may
not convey it at all. For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work. The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the
Program specifies that a certain numbered version of the GNU General
Public License "or any later version" applies to it, you have the
option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software
Foundation. If the Program does not specify a version number of the
GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs
If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

[Note that only a few files are distributed under this license.]

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.
1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based
on the Library, uncombined with any other library facilities,
conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it
is a work based on the Library, and explaining where to find the
accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions
of the GNU Lesser General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the
Library as you received it specifies that a certain numbered version
of the GNU Lesser General Public License "or any later version"
applies to it, you have the option of following the terms and
conditions either of that published version or of any later version
published by the Free Software Foundation. If the Library as you
received it does not specify a version number of the GNU Lesser
General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the
Library.

# COPYING.other
"*- org -*
#+TITLE: List of code with permissive licenses as used by GnuPG.
#+STARTUP: showall

* DNS resolver (dirmngr/dns.c)

dns.c - Recursive, Reentrant DNS Resolver.
---------------------------------------------------------------
Copyright (c) 2008, 2009, 2010, 2012-2016 William Ahern

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
persons to whom the Software is furnished to do so, subject to the
following conditions:

The above copyright notice and this permission notice shall be included
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

* TinySCHEME (tests/gpgscm/LICENSE.TinySCHEME)

Copyright (c) 2000, Dimitrios Souflis
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Dimitrios Souflis nor the names of the contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.209 commons/logging 1.2
1.209.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2006 The Apache Software Foundation.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
jar/org/apache/commons/logging/LogFactory.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2001-2004,2006 The Apache Software Foundation.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-
jar/org/apache/commons/logging/impl/AvalonLogger.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2004 The Apache Software Foundation.
 *
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

*/

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/WeakHashtable.java
No license file was found, but licenses were detected in source scan.

2004 The Apache Software Foundation.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/impl/package.html
* /opt/cola/permits/1135840457_1613613080.13/0/commons-logging-1-1-sources-9-jar/org/apache/commons/logging/package.html
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2001-2004 The Apache Software Foundation.
*
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

*/

* Copyright 2005 The Apache Software Foundation.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
1.210 audit 2.8.2-1ubuntu1.1
1.210.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.
For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)
"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.
(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.
You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms
of the ordinary General Public License).

To apply these terms, attach the following notices to the library.
It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.
10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the program
    `Gnomovision' (which makes passes at compilers) written by James Hacker.

    <signature of Ty Coon>, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.
1.211 libsigar 1.6.4
1.211.1 Available under license:
Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.212 threetenbp 1.4.4

1.212.1 Available under license:

Copyright (c) 2007-present, Stephen Colebourne & Michael Nascimento Santos.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of JSR-310 nor the names of its contributors
may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.213 slf4j-log4j 1.5.8

1.213.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.214 bouncy-castle 1.64

1.214.1 Available under license:

<html>
<body bgcolor=ffffff>

Copyright (c) 2000-2019 The Legion of the Bouncy Castle Inc. (http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The following organisations and people have contributed to the Bouncy Castle Cryptography Package.

Thanks, may your castles never deflate!

Donors

The following people donated financially to help with the release of 1.63:

Joshua Hight

The following people donated financially to help with the release of 1.61:

Bihari Babu

The following people donated financially to help with the release of 1.60:

Jens Neuhalfen and perillamint.

The following people donated financially to help with the release of 1.59:

Brian Reid.
The following people and organisations donated financially to help with the release of 1.57:
<br />
Joop Kaashoek and Pexus LLC.
</p>
<p>
We also wish to acknowledge financial and collaborative support from <a href="https://www.cisco.com/">CISCO</a> and additional financial support from <a href="https://www.primekey.com/">PrimeKey</a> towards developing the EST API for RFC 7030 support.
</p>
<p>
The following people and organisations donated financially to help with the release of 1.56:
<br />
DidiSoft, Cotiviti, Atanas Krachev, Encryptomatic LLC, LogicalAnswersIncSupporter
</p>
<p>
The following people and organisations donated financially to help with the release of 1.55:
<br />
Digistamp, RAM NAG
</p>
<p>
The following people and organisations donated financially to help with the release of 1.54:
<br />
Lobster GmbH
</p>
<p>
The following people and organisations donated financially to help with the release of 1.53:
<br />
Sheba, Ishmal Bartley, and Li-Chang Johnny Lo
</p>
<p>
The following people and organisations donated financially to help with the release of 1.52:
<br />
lecker@064buetterchen.de, Charles Proxy, Gunny Mills, Morgan Le Douget, Ben Whitaker, and Emilio Navarrete Lineros.
</p>
<p>
The following people and organisations donated financially to help with the release of 1.51:
<br />
Gup & Boz @ Alki Seattle, Bytemine Gmbh, Ted Pennings, Atanas Krachev, PrimeKey Solutions AB, Martin Paljak, CorseraFri19980116, CPU Terminator, Lindsay Bradford, kares, Philius, and Aaron Anderson.
</p>
<p>
Organisations
</p>
<ul>
<li>Holders of <a href="https://www.cryptoworkshop.com">Crypto Workshop Support Contracts</a>. Without the consulting time left over from support contracts being contributed back to working on the Bouncy Castle APIs, progress would be impossible. You know who you are!</li>
</ul>
Atlassian Software Systems donation of Confluence and JIRA licences.

Grier Forensics, for collaborating in the development of the S/MIME Toolkit and DANE SMIMEA functionality.

TU-Darmstadt, Computer Science Department, RBG, for the initial lightweight client side TLS implementation, which is based on MicroTLS and for help with qTESLA implementation. MicroTLS was developed by Erik Tews under the supervision of Dipl.-Ing. Henning Baer and Prof. Max Muehlhaeuser. qTESLA assistance was provided by Nina Bindel and Yinhua Xu.

TU-Darmstadt, Computer Science Department, RBG, for the initial Post Quantum provider, which was based on the FlexiProvider. The FlexiProvider was developed by the Theoretical Computer Science Research Group at TU-Darmstadt, Computer Science Department, RBG under the supervision of Prof. Dr. Johannes Buchmann. More information on the history of FlexiProvider can be found at:

Voxeo Labs - sponsorship of the initial development of APIs for DTLS 1.0 (RFC 4347), DTLS-SRTP key negotiation (RFC 5764), and server side TLS 1.1 (RFC 4346) and tested WebRTC compatibility. More information on Voxeo Labs can be found at

Voxeo Labs - sponsorship of the initial development of APIs for DTLS 1.0 (RFC 4347), DTLS-SRTP key negotiation (RFC 5764), and server side TLS 1.1 (RFC 4346) and tested WebRTC compatibility. More information on Voxeo Labs can be found at

Core Infrastructure Initiative - financial support towards developing the TLS API and JSSE provider that appeared in 1.56.

Additional CertPath testing and validation data from the CertPath testing tool developed by

Micro Focus - additional support towards further developing the TLS/DTLS API and the BCJSSE provider.

Tito Pena &lt;Fortunato.Pena@AdNovum.CH&gt; - initial RC5 implementation

Michael Lee &lt;yfl@mira.net&gt; - initial RC6 implementation, MD2 implementation

Nuno Santos &lt;nsantos@student.dei.uc.pt&gt; - finding toString bug in certificate object.

Brett Sealey &lt;bretts#mortbay.com&gt; - fixing the missing return problem in JDKKeyFactory (affected SSLeay private keys).

Victor A. Salaman &lt;salaman@teknos.com&gt; - fixing the bug in Cipher.java which caused it to ignore specified providers, fixing the bug in RSAKeyGenerator which caused keys to be occasionally produced 1 bit too small.

Eran Librach &lt;eranl#valicert.com&gt; - spotting and fixing the classLoader bug that occurs if javax.crypto and the provider aren't sharing the same classpath (occurs in JDK 1.3 and greater).

Jonathan Knudsen &lt;jonathan#LearningPatterns.com&gt; - porting information and restrictions when using the lightweight library with the MIDP environment.

Markus Niedermann &lt;markus.niedermann@softwired-inc.com&gt; - porting information and restrictions when using the lightweight library with the MIDP environment.

Mike Benham &lt;moxie@thoughtcrime.org&gt; - detection and fixing of an
incorrect weak key in the DES key generation support classes. Suggestions for simplifying DESedeParameter objects. Optimisations for the Blowfish engine and BufferedBlockCipher class.</li>
<li>Soren Hilmer &lt;hilmer@mail.tele.dk&gt; - initial implementation of netscape certificate request classes.</li>
<li>Uwe Guenther &lt;uwe@cscc.de&gt; - detection and fixing of 3 incorrect semi-weak keys in the DES key generation support classes.</li>
<li>Markus Bradtke &lt;mab@informatik.uni-kiel.de&gt; - fixing of a logic error in the JDKKeyStore class.</li>
<li>Waclaw Sierew &lt;waclaw.sierek@tpg.pl&gt; - fix to setOddParity in the DESParameter class. Assistance with adding ordering to X509 names for certificate generation, proper processing of byte strings in the ASN1 package, further simplifications and additional classes to improve pkcs7 support, bug fixes in CertPath API.</li>
<li>Ly-Na Phu &lt;lyna.phu@init-consulting.de&gt; - assistance in the addition of ISO 9796-1 padding.</li>
<li>Stefan K&ouml;nnl;psell &lt;k@k.uni-inf.tu-dresden.de&gt; - making the jdk 1.1 version of the collections API available. For further details see http://sourceforge.net/projects/jcf/.</li>
<li>Carmen Bastiaans &lt;cbastian@microbits.com.au&gt; - fixing the improper null pointer problem in the setting of certificates in the PKCS12 key store.</li>
<li>Tomas Gustavsson &lt;tommasg@primekey.se&gt; - initial implementation of the AuthorityInformationAccess, SubjectKeyIdentifier, AuthorityKeyIdentifier, CRLNumber, CRLReason, CertificatePolicies, V2TBSCertListGenerator, and X509V2CRLGenerator classes in the ASN.1 library. Additions to GeneralName class, other bug fixes in the X.509 package. Initial implementation of the CertificationRequest classes. getRevocationReason() patch for OCSP. Patch to SemanticsInformation to prevent ClassCastException.</li>
<li>Eugen Kuleshov &lt;euxx@hotmail.com&gt; - optimisations for Blowfish, assistance with PKCS12/keytool interoperability.</li>
<li>Megan Woods &lt;meganwoods@sekurafile.com&gt; - initial implementation of ECIES.</li>
<li>Christian Geuer-Pollmann &lt;geuerp@apache.org&gt; - adding IV's to the AESWrap implementations. Initial implementation of DESedeWrap.</li>
<li>Michael M&amp;#252;hle &lt;micnmouling.de&gt; - contributing the initial CertPath implementation and compatibility classes, fixing provider bug in JDK 1.1 java.security.cert.CertificateFactory compatibility class.</li>
<li>Michael Mansell &lt;m@michaelmansell.com&gt; - fixing the parsing of the empty DER set in theASN.1 library.</li>
<li>Eike Recker &lt;eike.recker@gmx.de&gt; - fixing misspelling of provider reference for RSA/1 and RSA/2.</li>
<li>Chris Southern &lt;CSouthern@baltimore.com&gt; - fixing misuse of specified provider in the PKCS10 certification request class.</li>
<li>Sidney Markowitz &lt;sidney@sidney.com&gt; - fixing null pointed exception on unknown OID in X509Name class, initial implementation of the three AES engines.</li>
<li>Chris Kerr &lt;ckerr@filonet.ca&gt; - initial implementation of the cms, asn1.cms, and the mail/smime packages, assistance in simplifying the ASN.1 package, miscellaneous other optimisations, NIST CertPath certification test, PKIXPolicyNode class, CertPath subtree validation and policy tree construction. We also
wish to acknowledge the generosity of Filonet
Corporation for allowing Chris to make the initial cms and mail/smime packages available to us.

Mike Bean &lt;mbean&#064;lucentradius.com&gt; - fixing the fall through bug in the IV algorithm parameters class.

Martin Petraschek &lt;m006;student.tuwien.ac.at&gt; - fixing ASN1 tagging so tag values up to 30 are now supported.

Jess Garms &lt;jgarms&#064;yahoo.com&gt; - fixing 112/168 key size bug for DESede key generation.

Mike Bremford &lt;mike&#064;big.faceless.org&gt; - contributing the initial PKCS7 implementation.

Shankar Srinivasan &lt;ss002&#064;yahoo.com&gt; - SMime interoperability testing and debugging.

Stef Hoeben &lt;ilstestef&#064;skynet.be&gt; - adding Montgomery multiplication to the BigInteger class.

Klaudiusz Ciosk &lt;kcioks&#064;max.com.pl&gt; - improving the compatibility of the SMIME package with the Sun JCE.

Thomas Houtekier &lt;Thomas.Houtekier&#064;tectrade.net&gt; - SMime testing and debugging. Interoperability with Biztalk.

Don Hillsberry &lt;hillsber&#064;dialcorp.com&gt; - SMime testing and debugging.

Kazuo Furuya &lt;kfuruya&#064;infoteria.co.jp&gt; - fixing root certificate chaining bug in PKCS12 key store.

Jdnovotny &lt;jdnovotny&#064;lbl.gov&gt; - initial work on the openSSL PEM processing.

Joel Hockey &lt;joel.hockey&#064;qsipayments.com&gt; - initial work on the openSSL PEM processing.

John Steenbruggen &lt;JohnS&#064;geotrust.com&gt; - fixing CertificationRequestInfo to handle cert request info objects without attribute blocks.

Justin Chapweske &lt;justin&#064;chaprzeske.com&gt; - ordering patch for Tiger message digest.

John Serock &lt;jsерock&#064;hotmail.com&gt; - fixing null pointer exception in constructor for ExtendedKeyUsage. Fixing of base OID bug in KeyPurposeId. Compliance of KeyUsage extension return value with security API.

Sascha Weinreuter &lt;Sascha.Weinreuter&#064;cit.de&gt; - fixed SMIME saveChanges() bug.

Andre Wehnert &lt;aw5&#064;mail.information.de&gt; - fixing key schedule problem in RC5-64, fixing buffer cleaning issue in buffered block cipher.

Luigi Lo Iacono &lt;lo_iacono&#064;nue.et-inf.uni-siegen.de&gt; - adding SIC mode to the blockciphers in the provider.

Tim Sakach &lt;tsakach&#064;certivo.net&gt; - SMIME v2 compatibility patches.

Marcus Povey &lt;mpovey&#064;brookes.ac.uk&gt; - adding the PGP mode to the lightweight API and the provider.

Sebastian Clau&szlig; &lt;sc2&#064;inf.tu-dresden.de&gt; - adding randomness setting to the certificate and CRL generators.

Nicolas Bielza &lt;nicolas.bielza&#064;alligacom.com&gt; - isolating the tagging bug in the ASN.1 library that was misrepresenting some ASN.1 constructed data types. Contributions to the streaming SMIME classes.

Casey Marshall &lt;rsdio&#064;metastatic.org&gt; - fixing the clone problem with Macs in the clean room JCE.

Rick Zeldes &lt;rick.zeldes&#064;eds.com&gt; - initial code for CMS/SMIME CompressedData.

Jarek Gawor &lt;gawor&#064;mcs.anl.gov&gt; - fixing ASN.1 sequence unpacking in BasicConstraints constructor.

Brett Neumeier &lt;random&#064;rmd.cx&gt; - patch to OriginatorIdentifierOrKey object, improvements to encoders package, introduction of UrlBase64.

Graham Coles &lt;graham.coles&#064;retail-logic.com&gt; - patch to isParityAdjusted in DESKeySpec.

J&ouml;rn von Kattchee &lt;J.Kattchee&#064;seeburger.de&gt; - patch to SMIMEGenerator for
preventing class cast exceptions with BodyParts containing Multipart objects.

- Matteo Artuso &lt;matartuso@libero.it&gt; - picking up the possible overead in ASN1InputStream.
- Julian Morrison &lt;julian@extropy.demon.co.uk&gt; - spotting the slow down in Diffie-Hellman key generation.
- Elmar Sonnenschein &lt;eso@esomail.de&gt; - fix to long conversion in clean room SecureRandom.
- Matteo Artuso &lt;matartuso@libero.it&gt; - Locale fix for the clean room JCE.
- Bryan Lovquist &lt;bkkl@#064:cps.com.au&gt; - Other provider compatibility fixes for CMS signing.
- Artem Portnoy &lt;Artem_Portnoy@ibi.com&gt; - generalisations for CMSProcessableBodyPart in S/MIME. Header fix for mime messages.
- Michael H&auml;usler &lt;haeusler@ponton-consulting.de&gt; - missing OID update for SHA1 with RSA Signature.
- Johan Seland &lt;johans#064:netfonds.no&gt; - general toString for BigInteger class.
- Johannes Nicolai &lt;johannes.nicolai@#064:novosec.com&gt; - further enhancements to OCSP response generation, fix to CertificateID issuer.
- Marc Doberva &lt;marc.doberva@#064:ilex-si.com&gt; - help in isolating the JSSE/BC RSA key issue.
- Jan Dvorak &lt;jan.dvorak@#064:mathan.cz&gt; - initial implementation of the light weight Null block cipher.
- Joe Cohen &lt;jcohen@#064:forums.sys.com&gt; - converting the ArrayOutOfBoundsException in DERInputSteam into what it should have been.
- Chris Long &lt;aclong@ece.cmu.edu&gt; - adding public key decoding to PEMReader.
- Hes Siemelink &lt;hes@#064:izecom.com&gt; - findIssuer fix for CertPathBuilder, toMimeMessage converter for Mail API, getSize() fix for zero length messages in SMIMEMessage.
- Stefan Puiu &lt;stefanpuiuro@#064:yahoo.com&gt; - initial implementation V3 policy mapping, policy qualifier objects in ASN.1 X.509 package.
- Kaiser Yang &lt;kaiseryang@#064:yahoo.com&gt; - Finding BigInteger loop problem in prime generation.
- Jiri Urbanec &lt;jiri.urbanec@#064:logicacmg.com&gt; - patch to fix defect in DERBMPString.equals().
- Justin Kolb &lt;jkolb@#064:pristx.com&gt; - patch to DSA signature generation in OpenPGP. Fix for the unexpected "Unexpected end of ZLIB input stream" exception.
- Ralf Hauser &lt;ralfhauser@#064:gmx.ch&gt; - patch to exception handling in PublicKeyRing, PEMReader, 1.4 build script, X509 Certificate Factory, CertPathValidatorUtilities, fromAddress null check in SignedMailValidator, ReadOnceInputStream testing in MIME tests.
- Michal Dvorak &lt;m_dvorak@#064:charta.cz&gt; - getNextUpdate patch for OCSP SingleResp.
- Klaus Greve Fiorentini &lt;klaus@#064:cpqd.com.br&gt; - array fix in PG PublicKeyEncSessionPacket.
- Olivier Refalo &lt;Olivier_Refalo@#064:ipf.com&gt; - null pointer exception fix for JDK 1.3 CMSSignedData objects.
- Mariusz Bandola &lt;mariusz.bandola@#064:cryptotech.com.pl&gt; - patch to DERGeneralizedTime. Compliance patch for OCSP TBSRequest class. Patch to X509Name for delaing with general objects in sequences.
- Brien Oberstein &lt;brien.oberstein@transacttools.net&gt; - patch to S2K algorithm in OpenPGP, initial PGP version 3 secret key support, initial PGP version 3 signature generation, RIPEMD160 addition to PGPUtility.
- Ian Haywood &lt;ian@#064:haywood.bpa.nu&gt; - addtion of getSignatureType to PGPSignature.
- Jonathan Edwards &lt;s34gull@#064:mac.com&gt; - initial support for reading multiple rings from a PGP key file.
- Andrew Thornton &lt;andrew@#064:caret.cam.ac.uk&gt; - patch for RSA PUBLIC KEY in PEMReader.
- Gregor Leander &lt;gl@#064:bos-bremen.de&gt; - initial parsing of multiple sequence entries in an X.500 Name.
- Antoon Bosselaers &lt;Antoon.Bosselaers@#064:esat.kuleuven.ac.be&gt; - help with RipeMD320
implementation.</li>
<li>Peter Sylvester &lt;Peter.Sylvester@064edelweb.fr&gt; - improvements to the ASN.1 BasicConstraints object.</li>
<li>Doug &lt;ummmmm@064myrealbox.com&gt; - addition of isEncryptionKey method to OpenPGP public keys.</li>
<li>Francois Staes &lt;fstaes@064netconsult.be&gt; - improvements to DEBitString, DERGeneralizedTime and initial implementation of DERGeneralString, addition of settable signed object info to CMSSignedDataGenerator, patch to DH key agreement.</li>
<li>W.R. Dittmer &lt;wdittmer@064cs.vu.nl&gt; - patch to decoding of SignatureCreationTime in BCPG. Patch to PGPKeyPair to fix nullpointer exception.</li>
<li>Perez Paz Luis Alberto &lt;laperez@064banxico.org.mx&gt; - patch to use of BitString in X.500 name.</li>
<li>James Wright &lt;James.Wright@064harte-hanks.com&gt; - patches for dealing with "odd" ArmoredInputStreams.</li>
<li>Jim Ford &lt;jim@064muirford.com&gt; - patch to PGPSecretKey to avoid null pointer exception on encoding secret keys, comments on KeyExpirationTime, getBitStrength for ElGamal keys. Signature creation time patch for newly created v4 signatures.</li>
<li>Michael Hausler &lt;hausler@064ponton-consulting.de&gt; - extra aliases for provider.</li>
<li>Sai Pullabhotla &lt;psai@064linoma.com&gt; - fix to PGPP compressed data generator to improve compression levels. Performance improvements for KeyBasedLargeFileProcessor.</li>
<li>Joseph Miller &lt;joseph@064digweb.net.nz&gt; - addition of ZeroBytePadding.</li>
<li>Lars &lt;lars@064sagemdenmark.dk&gt; - patch to explicit padded mode for CBC block cipher MAC.</li>
<li>Jeroen van Vianen &lt;jeroen@064vanvianen.nl&gt; - the Signed and Encrypted mail example.</li>
<li>Jun Sun &lt;JSun@064diversinet.com&gt; - patch to SecureRandom to work around problem in wtk 1.0.4 and wtk 2.1.</li>
<li>Petr Dukem &lt;pdukem@email.cz&gt; - patch to CMSSignedDataGenerator to allow it to work with PKCS11 providers.</li>
<li>Filipe Silva &lt;filipe.silva@064wedoconsulting.com&gt; - patch to fix overead issue in BCPGInputStream.</li>
<li>Alpesh Parmar &lt;alps@064linuxmail.org&gt; - patch for class cast problem in PGPPublicKey.getSignatures().</li>
<li>Jay Gengelbach &lt;jgengelbach@064webmethods.com&gt; - patch to fix isSigningKey in PGPSecretKey class, patch to hashedPackets in PGP signatureGenerator, initial cut for indefinite length output.</li>
<li>Doug &lt;doug@064tigerprivacy.com&gt; - public key ring patches for ElGamal Signatures, problem key ring data.</li>
<li>Matthew Mundy &lt;mmundy1@064umbc.edu&gt; - infinite loop prevention patch to PKCSS52ParametersGenerator.</li>
<li>Tom Cargill &lt;cargill@064proicon.com&gt; - spelling patch in provider.</li>
<li>Breitenstrom Christian &lt;C.Breitenstrom@064t-systems.com&gt; - compatibility patch to SignaturePacket, DetachedSignatureProcessor.</li>
<li>Zanotti Mirko &lt;zanotti@064cad.it&gt; - patch to ordered equality test for X509Name.</li>
<li>Nicola Scendoni &lt;nsendoni@064babelps.it&gt; - patch to add sorting to CertPath validation.</li>
<li>Ville Skytta &lt;auuml;@ville.skytta@064iki.fi&gt; - patch to CRLDistPoint for cRLIssuer field. KeyStore compliance on add patches. DiffieHellman patch for provider compliance. Support for PEM object "TRUSTED CERTIFICATE", Exception handling patch in PEMReader. JavaDoc clean up.</li>
<li>Bruce Gordon &lt;bruce.gordon@064savvis.net&gt; - patch to secret key creation encoding NullPointerException in OpenPGP, speed up for BCPGInputStream.</li>
<li>Miles Whiteley &lt;miles.whiteley@064savvis.net&gt; - "223" fix for BCPGInputStream new packets.</li>
<li>Albert Moliner &lt;amoliner@064evintia.com&gt; - initial TSP implementation.
Carlos Lozano &lt;carlos@evintia.com&gt; - initial TSP implementation, patch to SignerInformation for supporting repeated signers, initial updates for supporting repeated attributes in CMS.

Javier Delgadillo &lt;jaivi@codewarp.org&gt; - initial Mozilla PublicKeyAndChallenge classes.

Joni Hahkala &lt;joni.hahkala@cern.ch&gt; - initial implementations of VOMS Attribute Certificate Validation, letAttrSyntax, and ObjectDigestInfo. We also wish to thank the &lt;a href="http://www.eu-egee.org"&gt;EGEE project&lt;/a&gt; for making the work available.

Rolf Schillinger &lt;rolf@sir-wum.de&gt; - initial implementation of Attribute Certificate generation.

Sergey Bahtin &lt;Sergey.Bahtin@yahoo.com&gt; - fix for recovering certificate aliases in BKS and UBER key stores. Initial implementations of GOST-28147, GOST-3410, EC GOST-3410, GOST OFB mode (GOFB) and GOST-3411.

Franck Leroy &lt;Franck.Leroy@keynectis.com&gt; - ANS.1 set sorting. Contributions to TSP implementation. Test vectors for Bleichenbacher's forgery attack.

Atsuhiko Yamanaka &lt;ymnk@jcraft.com&gt; - patch for improving use of Montgomery numbers in BigInteger library. Patch to use size of private exponent in DH parameters.

Nickolay Bolshakov &lt;tyrex@reksoft.ru&gt; - patch for class cast exception in AuthorityInformationAccess class.

Soren Hilmer &lt;soren.hilmer@tietoenator.com&gt; - patches for CertID with issuerSerial set in TSP implementation, additional compliance testing.

Steve Mitchell &lt;mitchell@intertrust.com&gt; - patches for stateful path validator fix. Patch to allow BigInteger class to create negative numbers from byte arrays. Additions to allow different providers to be used for asymmetric/symmetric encryption in OpenPGP.

Optimisation to avoid redundant verification in path validator. Suggestion to use PKIXParameters.getSigProvider() correctly.

Dirk Eisner &lt;D.Eisner@seeburger.de&gt; - initial implementation of ISO 78164-4 padding.

Julien Pasquier &lt;julienpasquier@free.fr&gt; - initial implementation of attribute classes from RFC 3126. Fix to KEKIdentifier, OtherKeyAttribute parsing. Initial ContentHints class.

Matteo &lt;matartuso@libero.it&gt; - sequence patch to ASN1Dump.

Andrew Paterson &lt;andrew.paterson@burnsecs.com&gt; - patches to PGP tools, isRevoked method on PGPPublicKey.

Vladimir Molotkov &lt;vladimir.molotkov@intel.com&gt; - extensive provider exception handling compliance testing.

Florin Kollan &lt;adlocflo@web.de&gt; - fix to ElGamalKeyParameters equality testing.

Pavel Vassiliev &lt;paulvas@064gmail.com&gt; - Initial GOST28147Mac implementation.

Tom Pesman &lt;tom@064nxu.net&gt; - addition of DES-EDE encryption for RSAPrivate keys to PEMWriter.

Lukasz Kowalczyk &lt;lukaszkobowalczyk@064gmail.com&gt; - patch to fix parsing issue with OpenSSL PEM based certificate requests.

Arndt Hasch &lt;Arndt.Hasch@maxence.de&gt; - additional fix for partial reading with new style PGP packets.

Fix Bernd (KCDP 11) &lt;bernd.fixon@064credit-suisse.com&gt; - fix for 31 byte issue and exception throwing by Whirlpool.

David M. Lee &lt;dmlee@crossroads.com&gt; - code for add and remove secret key in the PGPSecretKeyRing class. Additions to S/MIME and CMS unit tests.

Mike Dillon &lt;md5@embody.org&gt; - additional checks for PGP secret and public key construction, patches to copyWithNewPassword.

tu-vi cung &lt;t2cung@hotmail.com&gt; - patch for out of bounds problem in getDecoderStream method.

Chris Schultz &lt;cschultz@064gmail.com&gt; - fix for InputStream constructor for
X509V2AttributeCertificate.

- David M. Lee &lt;dmlee@604crossroads.com&gt; - implementation assistance with streaming CMS classes.

- Joel Rees &lt;rees@604ddcom.co.jp&gt; - fix to correct getOID methods from returning same set on X.509 attribute certificates.

- Francesc Sau &lt;francesc.sau@604partners.netfocus.es&gt; - micro fix for tsp Accuracy class.

- Larry Bugbee &lt;lbugbee@606mac.com&gt; - initial ECNR implementation.

- Remi Blancher &lt;Remi.Blancher@604keynectis.com&gt; - Contributions to TSP implementation. Initial implementation of RFC 3739 and ICAO ASN.1 classes.

- Brian O'Rourke &lt;brianorourke@604gmail.com&gt; - patch for signature creation time override in OpenPGP.

- Andreas Schwier &lt;andreas.schwier@604cardcontact.de&gt; - initial implementation of ISO9797 MAC Algorithm 3, addition of DES-EDE 64 MAC to the provider, fix to EC point encoding, addition of EC and RSA-PSS OIDs to CMS, addition of AES-CMAC and DESede-CMAC to JCE provider.

- David Josse &lt;david.josse@604transacttools.net&gt; - Patch for trailer function in version 2 signature packets.

- Kishimoto Kazuhiko &lt;kazu-k@604hi-ho.ne.jp&gt; - RFC 3280 updates to policy processing in the CertPath validator. Additional test data not covered by NIST.

- Lawrence Tan &lt;lwrnctan@604gmail.com&gt; - Large field OID sample test data. Missing key types in JDKKeyFactory.

- Carlos Valiente &lt;superdupont@604gmail.com&gt; - Addition of CRL writing to the PEMWriter class.

- Keyon AG, Martin Christinat, &lt;a href="http://www.keyon.ch">http://www.keyon.ch&lt;/a&gt; - fixing incorrect ASN.1 encoding of field elements in X9FieldElement class.

- Olaf Keller, &lt;olaf.keller.bc@604bluewin.ch&gt; - initial implementation of the elliptic curves over binary fields F2m. Additional tests and modifications to elliptic curve support for both F2m and Fp. Performance improvements to F2m multiplication. Initial implementation of WNAF/WTNAF point multiplication. Improvement to k value generation in ECDSA.

- J&ouml;rg Eichhorn &lt;eichhorn@604ponton-consulting.de&gt; - patch to fix EOF read on SharedFileInputStream, support for F2m compression.

- Karsten Ohme &lt;widerstand@604t-online.de&gt; - initial check against for out of range data on non byte aligned RSA keys. Addition of equals/hashCode on ECCurve.Fp. Additional curve type support for Fp, contributions to F2m compression. F2m decoding for ECPiontUtil. Infinity fix and prime192v2 fix for Fp. Extra validation for RSA key creation. Fix to name typos for some OpenSSL key generators. RFC-1779 table, improved RFC 2253 compliance for X509Name. Additional constructor validation for X.509/ESS ASN.1 classes. Validation for Printable, IA5, and Numeric Strings. Fix for RFC 5280 NameConstraint checking for RDNs.

- Support for surrogate pairs in DERUTF8String, DER UTF8 test. Additional X.509 name attributes for ISIS-MTT, RFC 3039, addition of indirect CRL support, initial X509 LDAP CertStore implementation, CertificatePair class, and X509CertificatePair class. Contributions to X509Store/Parser infrastructure and design. CertPath support for implicit DSA parameters and a range of NameConstraints. Addition of support for V1 attribute certificates and attribute certificate path validation. Initial classes for ASN.1 ISIS-MTT support. Enhancements for improving compliance with the NIST CertPath tests.

- Carlos Lozano Ruiz &lt;carlos@604tradise.com&gt; - patch for &lt;ctrl&gt;&lt;m&gt; only handling in CRLOutputStream.

- John Alfred Prufrock &lt;j.a.prufrock@604gmail.com&gt; - mods to GOST-3411 and MD2 to support ExtendedDigest.

- Stefan Neusatz Guilhen &lt;sneusatz@604gmail.com&gt; - initial version of RoleSyntax, improvements to AttributeCertificateHolder and AttributeCertificateIssuer.

- Marzio Lo Giudice &lt;marzio.logiudice@604gmail.com&gt; - fix to endianess in KDF2BytesGenerator,
additional KDF2 tests.</li>
<li>Georg Lippold &lt;georg.lippold&#064gmx.de&gt; - initial implementation of NaccacheStern cipher.</li>
<li>Chris Viles &lt;chris_viles&#064yahoocom&gt; - fix to SignatureSubpacket critical bit setting.</li>
<li>Pasi Eronen &lt;Pasi.Eronen&#064nokia.com&gt; - extra toString() support for ASN.1 library. Initial patch for large OID components.</li>
<li>Maria Ivanova &lt;maria.ivanova&#064gmail.com&gt; - support for tags > 30 in ASN.1 parsing.</li>
<li>Armin H&auml;berling &lt;arminha&#064student.ethz.ch&gt; - first cut of internationalisation, initial PKIX validation classes.</li>
<li>Marius Schilder &lt;mschilder&#064gmail.com&gt; - main set of test vectors for Bleichenbacher's forgery attack.</li>
<li>Xavier Le Vournch &lt;xavier&#064brittanyswsoftware.com&gt; - general code clean ups.</li>
<li>Erik Tews &lt;e_tews&#064cdc.informatik.tu-darmstadt.de&gt; - initial threaded random seed generator, constant-time PKCS#1.5 decoding.</li>
<li>Thomas Dixon &lt;reikomusha&#064gmail.com&gt; - initial implementations of TEA/XTEA, Salsa20, ISAAC, and Noekeon. XTEA enhancements.</li>
<li>Frank Cornelis &lt;info&#064frankcornelis.be&gt; - addition of crlAccessMethod in X509ObjectIdentifiers.</li>
<li>Rui Joaquim &lt;rjovaquim&#064cc.isel.ipl.pt&gt; - initial implementation of RSA blinding for signatures.</li>
<li>David Stacey &lt;DStacey&#064allantgroup.com&gt; - addition of trust packet checking on revocation signatures in PGPSecretKeyRing.</li>
<li>Martijn Brinkers &lt;list&#06464mitm.nl&gt; - better exception handling in CMS enveloping. "just in time" modifications for CRL and Sequence evaluation.</li>
<li>Julius Davies &lt;juliusdavies&#064gmail.com&gt; - additional modes and algorithm support in PEMReader.</li>
<li>Matthias &lt;g&#064#064rttnr.de&gt; - GnuPG compatibility changes for PBEFileProcessor.</li>
<li>Olga K&auml;thler &lt;olga.kaethler&#064jhp-consulting.com&gt; - initial implementation of TeleTrusT EC curves, additional ISO 9797 MACs, contributions to EAC OIDs, addition of EAC algorithms to CMS Signing.</li>
<li>Germano Rizzo &lt;germano.rizzo&#064gmail.com&gt; - initial implementation of CMac, EAX, HC-128, and HC-256, optimisations for Salsa20.</li>
<li>N&amp;uuml;raicute;ria Mar&amp;iacute;cut;e; &lt;numaa&#064#064hotmail.com&gt; - patch for alternate data type recognition in CMSSignedDataParser.</li>
<li>Janis Schuller &lt;js&#064#064tzi.de&gt; - addition of NotationData packets for OpenPGP.</li>
<li>Michael Samblanet &lt;mike&#064#064samblanet.com&gt; - patches towards improved Sun/default provider support in CMS.</li>
<li>Mike StJohns &lt;mstjohns&#064#064comcast.net&gt; - patches for supporting empty subject in X.509 certificate generation, noneWithECDSA, updates to KeyPurposeId.</li>
<li>Ramon Keller &lt;ramon.keller&#064#064gmx.ch&gt; - patch to deal with null revocations return from other CRL in X509V2CRLEditor.</li>
<li>Mark Nelson &lt;mark&#064#064nbr.com&gt; - correction to excluded DN in name constraints processing for PKIX processing.</li>
<li>Eugene Golushkov &lt;eugene_gff&#064ukr.net&gt; - mask fix to single byte read in TlsInputStream.</li>
<li>Julien Pasquier &lt;julienpasquier&#064free.fr&gt; - additional classes for supporting signature policy and signer certificates in the ASN.1 esf and ess libraries.
Peter Knopp &lt;pknopp@mtg.de&gt; - fix for named curve recognition in ECGOST key generation.

Jakub Gwozdz &lt;gwozdziu@#064rp.pl&gt; - addition of getTs() to TimeStampTokenInfo.

Bartosz Malkowski &lt;bmalkow@#064tigase.org&gt; - initial implementation of VMPC cipher, VMPCRandomGenerator, VMPCMac.

Tal Yacobi &lt;tal.yacobi@#064octavian-tech.com&gt; - fix for issue in OpenPGP examples [#BJA-55]

Massimiliano Ziccardi &lt;massimiliano.ziccardi@#064gmail.comt&gt; - support for counter signature reading in CMS API, update for multiple counter signature attributes, JCA compliance patch for PEM parsing in CertificateFactory.

Andrey Pavlenko &lt;andrey.a.pavlenko@#064gmail.com&gt; - security manager patch for PKCS1Encoding property check.

J Ross Nicoll &lt;jrn@#064jrn.me.uk&gt; - improved exception handling for getInstance() in ASN.1 library.

Matthew Stevenson &lt;mavricknz@#064yahoo.com&gt; - patch to constructor for CRMF CertSequence.

Gabriele Contini &lt;gcontini@#064hotpop.com&gt; - identified a bug in ASN.1 library with handling of unterminated NDFEs.

Roelof Naude &lt;roelof.naude@#064epiuse.com&gt; - patch for TLS client to send empty client certs in response to HP_CERTIFICATE_REQUEST.

Patrick Peck &lt;peck@#064signature.at&gt; - identified problem with DERApplicationSpecific and high tag numbers in ASN.1 library.

Michael LeMay &lt;lemaymd@#064lemaymd.com&gt; - identified problem with EAX [#BJA-93].

Alex Dupre &lt;ale@#064FreeBSD.org&gt; - fix to use of Signature rather than SignatureSpi in provider [#BJA-90]. Addition of null provider use to SignedPublicKeyAndChallenge and PKCS10CertificationRequest

Michael Schoene &lt;michael@#064sgrid-und-michael.de&gt; - fix of improper handling of null in ExtendedPKIXParameters.setTrustedACIssuers(), check for V2 signingCertificate attribute in TimeStampResponse.validate().

Ion Larrañaga &lt;ilarra@#064s21sec.com&gt; - fix to default partial packet generation in BCPGOutputStream.


Stefan Meyer &lt;stefan.meyer@#064ewe.de&gt; - backport for PKIXCertPathValidator and SMIMESignedMailReviewer.

Robert J. Moore &lt;Robert.J.Moore@#064allanbank.com&gt; - speedups for OpenPGPCFB mode, clean room JCE patches.

Rui Hodai &lt;rui@#064po.ntts.co.jp&gt; - speed ups for Camellia implementation, CamelliaLightEngine.

Emir Bucalovic &lt;emir.bucalovic@#064mail.com&gt; - initial implementation of Grain-v1 and Grain-128.

Torbjorn Svensson &lt;tobbe79@#064gmail.com&gt; - initial implementation of Grain-v1 and Grain-128.

Paul FitzPatrick &lt;bouncycastle_pfitz@#064fitzpatrick.cc&gt; - error message fix to X509LDAPCertStoreSpi, comparison fix to BCStrictStyle.

Henrik Andersson &lt;k.henrik.andersson@#064gmail.com&gt; - addition of UniqueIssuerID to certificate generation.

Cagdas Cirit &lt;cagdascirit@#064gmail.com&gt; - subjectAlternativeName fix for x.509CertStoreSelector.

Harakiri &lt;harakiri_23@#064yahoo.com&gt; - datahandler patch for attached parts in SMIME signatures.

Pedro Henriques &lt;pmahenriques@#064gmail.com&gt; - explicit bounds checking for DESKeyGenerator, code simplification for OAEPEncoding.

Lothar Kimmeringer &lt;job@#064kiimmering.de&gt; - verbose mode for ASN1Dump, support for DERExternal, DNS performance fix for SMIME API.

Richard Farr &lt;r.farr.se@#064gmail.com&gt; - initial SRP-6a implementation.

Thomas Castiglione &lt;tcastiglione@#064au.ibm.com&gt; - patch to encoding for CRMF OptionalValidity.
Elisabetta Romani &lt;eromani@sogei.it&gt; patch for recognising multiple counter signatures.

Robin Lundgren &lt;r737lundgren#64gmail.com&gt; CMPCertificate constructor from X509CertificateStructure fix.

Petr Kadlec &lt;mormegil#64centrum.cz&gt; fix to sign extension key and IV problem in HC-128, HC-256.

Andreas Antener &lt;antener_a@#64gmx.ch&gt; fix to buffer reset in AsymmetricBufferedBlockCipher.

Harendra Rawat &lt;hsrawat#64yahoo.com&gt; fix for BERConstructedOctetString.

Rolf Lindemann &lt;lindemann#64trustcenter.de&gt; patch for PKCS12 key store to support more flexible attribute specifications [#BMA-42].

Alex Artamonov &lt;alexart.home&64gmail.com&gt; name look up patch for GOST-2001 parameters.

Mike Lyons &lt;mlyons#64layer7tech.com&gt; work arounds for EC JDK bug 6738532 and JSSE EC naming conventions.

Chris Cole &lt;chris_h_cole#64yahoo.com&gt; identified a problem handling null passwords when loading a BKS keystore.

Tomas Krivanek &lt;tom@#64atack.cz&gt; added checking of Sender header to SignedMailValidator.

Michael &lt;emfau@#64t-online.de&gt; correction of field error in getResponse method in CertRepMessage.

Trevor Perrin &lt;trevor#64cryptography.com&gt; addition of constant time equals to avoid possible timing attacks.

Markus Kilaringus &lt;markus#64primekey.se&gt; several enhancements to TimeStampResponseGenerator.

Dario Novakovic &lt;darionis#64yahoo.com&gt; fix for NPE when checking revocation reason on CRL without extensions.

Michael Smith &lt;msmith#64cbnco.com&gt; bug fixes and enhancements to the CMP and CRMF classes, initial Master List classes.

Andrea Zilio &lt;andrea.zilio@#64gmail.com&gt; fix for PEM password encryption of private keys.

Alex Birkett &lt;alex@#64birkett.co.uk&gt; added support for EC cipher suites in TLS client (RFC 4492) [#BJA-291].

Wayne Grant &lt;waynedgrant#64yahoo.com&gt; additional OIDs for PCKS10 and certificate generation support.

Frank Cornelis &lt;info@#64frankcornelis.be&gt; additional support classes for CAdES, enhancements to OCSP classes.

Jan Dittberner &lt;jan@#64dittberner.info&gt; addHeader patch for SMIME generator.

Bob McGowan &lt;boab.mcgoo@btinternet.com&gt; addHeader patch for SMIME generator.

Ivo Matheis &lt;i.matheis@seeburger.de&gt; fix to padding verification in ISO-9796-1.

Marco Sandrini &lt;essche@#64gmail.com&gt; patch to add IV to ISO9797Alg3Mac.

Alf Malf &lt;alfilmalf#64hotmail.com&gt; removal of unnecessary limit in CMSContentInfoParser.

Alfonso Massa &lt;alfonso.massa@#64insiel.it&gt; contributions to CMS time stamp classes.

Giacomo Boccanaro &lt;gboccaro@#64unimaticaspa.it&gt; initial work on CMSTimeStampedDataParser.

Arnis Tartu &lt;arnis@#64ut.ee&gt; patches for dealing with OIDs with specific key sizes associated in CMS.

Janusz Sikociski &lt;j.Sikocinski@#64gdiez.pl&gt; addition of Features subpacket support to OpenPGP API.

Juri Hudolejev &lt;jhudolejev#64gmail.com&gt; JavaDoc fix to CMSSignedDataParser.

Liane Velten &lt;liane.velten@#64hjpc-consulting.com&gt; fine tuning of code for DHParameters validation.
<li>Shawn Willden &lt;swillden#064google.com&gt; additional functionality to PGPKeyRing.</li><li>Atanas Krachev &lt;akrachev#064gmail.com&gt; added support for revocation signatures in OpenPGP.</li><li>Mickael Laiking &lt;mickael.laiking#064keynectis.com&gt; initial cut of EAC classes.</li><li>Tim Buktu &lt;tbuktu#064hotmail.com&gt; Initial implementation of NTRU signing and encryption.</li><li>Bernd &lt;rbernd#064gmail.com&gt; Fix for open of PGP literal data stream with UTF-8 naming.</li><li>Steing Inge Morisbak &lt;stein.inge.morisbak#064BEKK.no&gt; Test code for lower case Hex data in PEM headers.</li><li>Andreas Schmid &lt;andreas.schmid#064tngtech.com&gt; Additional expiry time check in PGPPublicKeys.</li><li>Phil Steitz &lt;phil.steitz#064gmail.com&gt; Final patch eliminating JCE dependencies in the OpenPGP BC classes.</li><li>Ignat Korchagin &lt;ignat.korchagin#064gmail.com&gt; Initial implementation of DSTU-4145-2002, long hash fix for DSTU-4145-2002.</li><li>Petar Petrov &lt;p.petrov#064bers-soft.com&gt; Testing and debugging of UTF-8 OpenPGP passwords.</li><li>Daniel Fitzpatrick &lt;daniel.f.nwr#064gmail.com&gt; Initial implementation of ephemeral key support for IES, initial implementations of RSA-KEM and ECIES-KEM, initial implementation of homogeneous projective coordinates for EC.</li><li>Andy Neilson &lt;Andy.Neilson#quest.com&gt; a further patches to deal with multiple providers and PEMReader.</li><li>Ted Shaw &lt;xiao.xj#064gmail.com&gt; patch to MiscPEMGenerator for handling new PKCS10CertificationRequests.</li><li>Eleriseth &lt;Eleriseth#WPECGLtYbVi8RI6Y7Vz1Lvd2EUVW99v3yNV3IWROG8.fms&gt; speed up for SIC/CTR mode. Provider compatibility generalisations for EC operations.</li><li>Kenny Root &lt;kenny#064the-b.org&gt; patch for issuerAltName, subjectAltName support in X509CertificateObject, BaseBlockCipher.getIV() patch for AEAD.</li><li>Maarten Bodewes &lt;maarten.bodewes#064gmail.com&gt; initial implementation of HKDF and NIST SP 800-108 MAC based KDF functions.</li><li>Philip Clay &lt;phil_b#064yahoo.com&gt; Initial implementation of J-PAKE.</li><li>Brian Carlstrom &lt;bdc#064carlstrom.com&gt; compliance patches for some JCA/JCE keystore and cipher classes, miscellaneous code quality improvements, intial provider PBKDF2WithHmacSHA1 SecretKeyFactory.</li><li>Samuel Lid&eacute;n Borell &lt;samuel#064primekey.se&gt; patch to add DSTU-4145 to DefaultSignatureAlgorithmFinder</li><li>Sergio Demian Lerner &lt;sergiolerner#064certimix.com&gt; pointing out isInfinity issue in ECDSASigner signature verification.</li><li>Tim Whittington &lt;Tim.Whittington#064orionhealth.com&gt; patch to remove extra init call in CMac, additional of Memorable interface for Digest classes, initial implementation of GMAC, further correctness tests for IV and reset processing in OCB, CCM, and block cipher reset. Initial implementation of Skein, XSalsa20, ChaCha, reduced round Salsa20, Threefish, and the Poly1305 MAC. Documentation updates. Added OCB support to Noekeon and CAST6 in the provider, exception testing for CTS, optimisations for CCM, provider support for AAD cipher methods, safe CipherInput/OutputStream implementations for use with AAD and subsequent bug fixes, cleanup after IDEA patent expiry, win on JCE SipHash support, optimisations for AESFastEngine, further work on EncodableDigest for SHA-2 digests, contributions to BCrypt/OpenBSD BCrypt, PGP API documentation and code quality work.</li><li>Marcus Lundblad &lt;marcus.lundblad#064primekey.se&gt; patch for working around JDK jarsigner TSP bug, optional setting of IssuerSerial in TimeStampTokenGenerator, additional extensions enhancement for time stamp token generation.</li><li>Andrey Zhozhin &lt;zhozhin#xrm.ru&gt; patch for override of TSP SignerInfo attributes.
Sergey Tiunov &lt;t5555d&#064;gmail.com&gt; initial cut of DVCS classes.

Damian Kolasa &lt;tfatredyy&#064;gmail.com&gt; ASN1Sequence patch for class cast issue in X9Curve.

Ash Hughes &lt;ashley.hughes&#064;blueyonder.co.uk&gt; patches for supporting PGPSecretKeyRing/PGPSecretKeys encodings with empty private keys, initial code for PGPSignatureSubpacketVector.getEmbeddedSignatures().

Daniel Hirscher &lt;dev&#064;064daniel-hirscher.de&gt; patch to support parsing of explicit EC parameters in PEM files.

Daniele Ricci &lt;daniele.athome&#064;064gmail.com&gt; initial implementation of EC keys for OpenPGP and RFC6637 support.

Matti Aarnio &lt;matti.aarnio&#064;methics.fi&gt; tweaks to any build to remove dependence on shell scripts. Initial SM3 digest implementation, some EC related code cleanups, JavaDoc improvements for ASN.1 classes, addition of NONEwithRSA to lightweight RSADigestSigner.

Babak Najafi &lt;bnajafi&#064;akamai.com&gt; fixes to OpenPGP NotationData to prevent truncation problems.

Eric Müller &lt;eric.mueller&#064;sage.de&gt; additional standard algorithm name lookups in JcaPEMKeyConverter.

Mathias Herberts &lt;Mathias.Herberts&#064;064gmail.com&gt; fix to inOff usage in RFC3394WrapEngine.

Daniil Ivanov &lt;daniil.ivanov&#064;064gmail.com&gt; addition of provider support for GOST HMAC SecretKeyFactory.

Daniele Grasso &lt;daniele.grasso86&#064;064gmail.com&gt; contributions to final Key calculation code for SRP6.

Andrey Utkin &lt;cindrhc&#064;064gmail.com&gt; patch to reconstruction of ECGOST keys from PrivateKeyInfo objects in provider classes.

Arnis Tartu &lt;arnis&#064;ut.ee&gt; checker for generated key vs OID in JceCMSContentEncryptorBuilder.

AxelVDB &lt;axel-vdb&#064;riseup.net&gt; initial implementation of Shacal2.

Roberto Tyley &lt;rtyley@064gmail.com&gt; further work on completing gradle build.

Waldemar Dick &lt;wdick@064devmue.de&gt; code improvement in x500 ASN.1 package.

Sid Steward &lt;sid.steward@064pdflabs.com&gt; code improvements to ASN1Boolean.

Alex Klyubin &lt;klyubin@064google.com&gt; AlgorithmParameters check for EC key agreement.

Jonathan Gillett &lt;gsoc.student@064gmail.com&gt; Initial support for block cipher IVs in IESEngine, IES MAC length check bug fix.

Andreas Reiter &lt;andreas.reiter@064aik.tugraz.at&gt; Reported incomplete status of CertificateVerify processing in (D)TLS server, and provided fix.

Kieran Miller &lt;kieran.miller@064gmail.com&gt; initial implementation for RFC 5649 key wrap with padding.

Oliver Ehli &lt;ehli@064arago.de&gt; Additional support for BSI plain ECDSA in the provider.

Daniel Heldt &lt;Daniel.Heldt@064cryptovision.com&gt; Initial support for encodable state message digests.

Robert Bushman &lt;bouncycastle@064traxel.com&gt; Clean up of DirectKeySignature example.

Maurice Aarts &lt;aarts@064ricsure.com&gt; updated to KDF generator to follow NIST SP 800-108.

Franziskus Kiefer &lt;https://github.com/franziskuskiefer&gt; initial implementation of Cramer-Shoup.

KB Sriram &lt;mail_kb@yahoo.com&gt; testing for odd encodings for PGP User Attribute Subpackets.

Marco Schulze &lt;marco@064nightlabs.de&gt; Reported verification bug in GenericSigner.

Martin Schaef &lt;https://github.com/martinschaef&gt; contributed a code-cleanup patch.

dstutz &lt;https://github.com/dstutz&gt; added iteration count setters to PKCS#12 PBE mac/key generator builders.
Tobias Wich &lt;tobias.wich@064ecsec.de&gt;: Provided patch for TLS to work around servers sending supported elliptic curves extension unexpectedly.

Hauke Mehrtens &lt;hauke@064hauke-m.de&gt;: TLS patch to add ECDHE\_ECDSA \_CCM ciphersuites from RFC 7251.

Daniel Zimmerman &lt;dmz@064galois.com&gt;: Further key quality improvements to RSAKeyPairGenerator.

Jens Kapitza &lt;lj.kapitza@064schwarze-allianz.de&gt;: Iterable support in OpenPGP API, code cleanup in OpenPGP API.

Johan Eklund &lt;johan@064primekey.se&gt;: Update to RFC 6960 for OCSP\_ObjectIdentifiers.

nikosn &lt;https://github.com/nikosn&gt;: Fix to encoding of EC private keys to ensure encoding matches order length.

Axel von dem Bruch &lt;axel-vdb@064riseup.net&gt;: Contributions to BCrypt/OpenBSD\_BCrypt, original version of Blake2b\_Digest.

Derek Atkins &lt;derek@064ihtfp.com&gt;: Documentation fixes to X9\_ObjectIdentifiers.

Peter Jr Halicky &lt;peto@064halicky.sk&gt;: Correction to notification/error message handling in SignedMailValidator.


Thomas Belot &lt;thomas.belot+BC@gmail.com&gt;: Initial CertPathLoopTest for demonstrating stack overflow issue.

Rich DiCroce &lt;https://github.com/rdcroce&gt;: Initial implementation of server-side TLS\_SRP support. TLS API extension to support non-blocking usage.

Bj\_ouml;rn Kautler &lt;https://github.com/Vampire&gt;: Refinements to cert path validation (authority key addition, certificate order preservation).

Dominik Sch\_uuml;mann &lt;https://github.com/dschuermann&gt;: Method for returning signatures/verifications without user IDs on PG\_Public\_Key, method for exposing S2K in PG\_Secret\_Key, constants for GNU protection modes in S2K classes, optional version header for armored output.

Michael &lt;MSKnete@halicky.sk&gt;: Initial fix for bitStrength issue for OpenPGP EC keys, correction for generic type on Recipient\_Information\_Store.

Tobias Wagner &lt;tobias.wagner@064n-design.de&gt;: Fix SecureRandom handling in Bc\_Asymmetric\_Key\_Wrapper [\#BJA-536].

Sergio Giro &lt;sgiro@google.com&gt;: Fixed adding of additional stores from CRL distribution point [\#BJA-537]. Fixed missing null check for CRL certificate issuer [\#BJA-538], removal of risky zeroisation code in PBE\_Java, check for salt in PBE\_Ke y\_s that require it.

bschuette &lt;https://github.com/bschuette&gt;: Fixed typo in Default\_Signature\_Algorithm\_Identifier\_Finder, additional methods on CMS\_Signed\_Data\_Parser.

Leonard Dallot &lt;https://github.com/dallotTazTag&gt;: Fix to S2K usage of none on changing passwords on keys without passwords originally.

Jan Willem Janssen &lt;j.w.janssen+bouncycastle@064lxtemple.nl&gt;: Support for D\_S\_Parameters in lightweight Subject\_Public\_Key\_Info\_Factory, initial object signer verifier for BC lightweight EC.

Sebastian Oerding &lt;sebastian.oerding@robotron.de&gt;: Fixes to toString() in x509\_Certificate\_Policies.

Kai Kramer &lt;kai.kramer@064gmail.com&gt;: Code to deal with orphaned chain certificates in the PKCS\#12 Key\_Store.

Benoit Charles &lt;benoit.charles@opentrust.com&gt;: Fix for IES data length check on decryption.

Niko &lt;nfink95@064gmail.com&gt;: Fix to cast issue in get\_Output\_Size() for EC\_IES.

akwizgran &lt;https://github.com/akwizgran&gt;: Fixed clone of key in Blake\_2\_Digest copy constructor, Blake2b reset issue for variant keys.

Matthias Edelhoff &lt;Matthias.Edelhoff@cryptovision.com&gt;: Basic\_Constraints\_Validation pathlen fix in PK\_IX\_Cert\_Path\_Classes.
Lukasz Deputat &lt;lukasz.deputat@gmail.com&gt; Fixed bugs in TlsUtils read methods [#BJA-592].

Justin Ludwig &lt;https://github.com/justinludwig&gt; Iterator fix for PGPObjFactory to handle stream packets at start of iterated data.

Andrzejacutte Berenguel &lt;https://github.com/berenguel&gt; Fix to include ECNamedCurveSpec in EC AlgorithmParameterSpi.

Sławomir Jaranowski &lt;https://github.com/slawekjaranowski&gt; Patch to make cipher/hash/signature name methods in PGP internal API public.

Andrey Vasilyev &lt;https://github.com/andrey-vasilyev&gt; Initial implementation of GOST R 34.11-2012.

William Glanton &lt;wglanton77@gmail.com&gt; Fixed bug in Poly1305 [#BJA-620].

jdvorak001 &lt;https://github.com/jdvorak001&gt; Speed improvements for ASN.1 ObjectIdentifier cache.

Joseph Naegle &lt;jnaegle@64grierforensics.com&gt; Patch for handling multiple certificates in a DANE SMIMEA entry.


The Google Security Team (Project Wycheproof) &lt;https://github.com/google/wycheproof&gt; defect analysis and additional test cases for the provider.

Gorka Irazoqui &lt;girazoki@wpi.edu&gt; from Intel Security Center of Excellence &lt;https://security-center.intel.com/&gt; detection of the issue with AESFastEngine (CVE-2016-1000339), additional suggestions for improvement to hardening of AESEngine and finding cache sensitivities in EC key generation/signing.

Joerg Senekowitsch &lt;joerg.senekowitsch@veridos.com&gt; patch to deal with hard coded boolean in EAC ECDSAPublicKey.

Alexandr Krivoshta &lt;wipe@064ya.ru&gt; N4 calculation fix to GOBF mode.

Artem Storozhuk &lt;storozh72@gmail.com&gt; N4 calculation fix to GOBF mode.

Na Yu &lt;na.yu@064samsung.com&gt; Constructor patches to CMC PKIData.

Evangelos Karatsiolis &lt;ekaratsiolis@064mtg.de&gt; Corrected use of explicit tagging in X.509 PolicyConstraints class.

VivleSoren &lt;https://github.com/VivleSoren&gt; additional constructor for McElieceCA2PrivateKeyParameters.

mtausig &lt;https://github.com/mtausig&gt; JavaDoc fix for MCSEncryptedDataGenerator.


Sebastian Wolfgang Roland &lt;sebastianwolfgang.roland@stud.tu-darmstadt.de&gt; Initial XMSS/XMSS-MT implementation.

didisoft &lt;https://github.com/didisoft&gt; test code for PGP signature removal involving user ids.

Mike Safonov &lt;https://github.com/MikeSafonov&gt; initial implementation of GOST3410-2012 for lightweight provider and JCA, parameters patches for ECGOST keys, initial implementation of GOST3412-2015, addition of fromExtensions() for CRLDistPoint.

Artem Storozhuk &lt;storozh72@gmail.com&gt; initial implementation of DSTU7564 (digest) and DSTU7624 (cipher) and their associated modes.

Andreas Glaser &lt;andreas.glaser@064gi.de.com&gt; patch to recognise ANSSI curves for PKCS#10 requests.

codeborne &lt;https://github.com/cbxp&gt; patch to correct OIDs used in public key digest parameters for ECGOST-2012.

FauxFaux &lt;https://github.com/FauxFaux&gt; patch for JDK 1.9 update to DRBG.java.

4garbage &lt;https://github.com/4garbage&gt; patch to allow GOST3410-94 private keys encoded as integers.

ekszz &lt;https://github.com/ekszz&gt; corrections to SM2 signer to include default identity value.

jminer &lt;https://github.com/jminer&gt; fix to Blake2b for hashes in range of 2**64-127 to 2**64.
<li>str4d &lt;https://github.com/str4d&gt; initial implementation of Blake2s</li>
<li>Scott Woodward &lt;scott@064bit3consulting.com&gt; performance fixes for CTRSP800DRBG.</li>
<li>David Straw &lt;https://github.com/isomarcte&gt; fix for off by one error in SCRYPT bounds checking.</li>
<li>Chris Mccown &lt;0xchristsmccown@#064gmail.com&gt; identification of serialisation issue with XMSS/XMSSMT private keys (see also CVE-2018-1000613).</li>
<li>ZZMarquis &lt;https://github.com/ZZMarquis&gt; offset patches for SM2 encryption and decryption, improvement to Array constant time comparison.</li>
<li>Andreas Kreutschmer &lt;https://github.com/Akretsch&gt; NPE fix for CertTemplate.getVersion().</li>
<li>Armin Lunkeit, Michael Tautenhahn &lt;https://github.com/isomarcte&gt; fix for off by one error in SCRYPT bounds checking.</li>
<li>Chris Mccown &lt;0xchristsmccown@#064gmail.com&gt; identification of serialisation issue with XMSS/XMSSMT private keys (see also CVE-2018-1000613).</li>
<li>ZZMarquis &lt;https://github.com/ZZMarquis&gt; offset patches for SM2 encryption and decryption, improvement to Array constant time comparison.</li>
<li>Andreas Kreutschmer &lt;https://github.com/Akretsch&gt; NPE fix for CertTemplate.getVersion().</li>
<li>Armin Lunkeit, Michael Tautenhahn &lt;https://github.com/isomarcte&gt; fix for off by one error in SCRYPT bounds checking.</li>
<li>Chris Mccown &lt;0xchristsmccown@#064gmail.com&gt; identification of serialisation issue with XMSS/XMSSMT private keys (see also CVE-2018-1000613).</li>
<li>ZZMarquis &lt;https://github.com/ZZMarquis&gt; offset patches for SM2 encryption and decryption, improvement to Array constant time comparison.</li>
<li>Andreas Kreutschmer &lt;https://github.com/Akretsch&gt; NPE fix for CertTemplate.getVersion().</li>
<li>Armin Lunkeit, Michael Tautenhahn &lt;https://github.com/isomarcte&gt; fix for off by one error in SCRYPT bounds checking.</li>
<li>Chris Mccown &lt;0xchristsmccown@#064gmail.com&gt; identification of serialisation issue with XMSS/XMSSMT private keys (see also CVE-2018-1000613).</li>
<li>ZZMarquis &lt;https://github.com/ZZMarquis&gt; offset patches for SM2 encryption and decryption, improvement to Array constant time comparison.
René Korthaus &lt;https://github.com/securitykernel&gt; Update to XMSS/XMSS^MT OID values to bring them in line with RFC 8391.

THausherr &lt;https://github.com/THausherr&gt; Addition generic support for CMS/TSP functions.

Gaylor Bosson &lt;https://github.com/Gilthoniel&gt; Initial implementation of Blake2xs.

gaelalaire &lt;https://github.com/gaelalaire&gt; Patch for unprotected PGP private keys as SExpr.

im-scooter &lt;https://github.com/im-scooter&gt; Patch for case-insensitive behaviour of Param-Z.

Nick hitchan &lt;https://github.com/hitchan&gt; Fix for typo in engineInitSign() in EdEc SignatureSpi.

dbusche &lt;https://github.com/dbusche&gt; Argon2 optimisations.

Daniel Heldt &lt;https://github.com/dheldt&gt; Fixing encodings in unicode tests to allow a wider range of Java compilers to work. Tweak to inheritance in JceKeyAgreeRecipient.

Ugochukwu Mmaduekwe &lt;https://github.com/Xor-el&gt; Fix for initially bugged legacy Integers.numberOfLeadingZeros method.

gsealy &lt;https://github.com/Gsealy&gt; Addition of PBKDF2withHmacSM3 to the provider.

aphuang2013 &lt;https://github.com/aphuang2013&gt; Update to path validation in EST service for ClearPath EST.

Kevin Herron &lt;https://github.com/kevinherron&gt; Initial ChaCha20Poly1305 prototype.

vkreml &lt;https://github.com/svkreml&gt; GOST compliance change for DefaultCMSSignatureEncryptionAlgorithmFinder.

Tobias Ospelt &lt;tobias@pentagrid.ch&gt; Identification of 1.63 regression in ASN.1 parsing.

aphuang2013 &lt;https://github.com/aphuang2013&gt; Update to path validation in EST service for ClearPath EST.

Kevin Herron &lt;https://github.com/kevinherron&gt; Initial ChaCha20Poly1305 prototype.

vkreml &lt;https://github.com/svkreml&gt; GOST compliance change for DefaultCMSSignatureEncryptionAlgorithmFinder.

Tobias Ospelt &lt;tobias@pentagrid.ch&gt; Identification of 1.63 regression in ASN.1 parsing.

1.215 hamcrest 1.3

1.215.1 Available under license:

1.215.1.1 Available under license:
under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be
covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder. </font></p></ul>

<ul><font size="2">

<font size="2">c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.</font></ul>

<font size="2">d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement. </font></ul>

<font size="2">3. REQUIREMENTS</font><p></p>

<font size="2">A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:</font>

<font size="2">a) it complies with the terms and conditions of this Agreement; and</font></ul>

<font size="2">b) its license agreement:</font>

<font size="2">i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose; </font></ul>

<font size="2">ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits; </font></ul>

<font size="2">iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and</font></ul>
iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.
WARRANTY</B></FONT></P><FONT SIZE="2">EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is</FONT><FONT SIZE="2"> solely responsible for determining the appropriateness of using and distributing </FONT><FONT SIZE="2">the Program</FONT><FONT SIZE="2"> and assumes all risks associated with its exercise of rights under this Agreement</FONT><FONT SIZE="2">, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, </FONT><FONT SIZE="2">programs or equipment, and unavailability or interruption of operations</FONT><FONT SIZE="2">. </FONT><FONT SIZE="2"></FONT><FONT SIZE="2"></FONT><FONT SIZE="2">EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTOR SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.</FONT><FONT SIZE="2">.</FONT><P><FONT SIZE="2"></FONT></P><P><FONT SIZE="2">6. DISCLAIMER OF LIABILITY</FONT></P><P><FONT SIZE="2">EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTOR SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.</FONT></P><P><FONT SIZE="2"></FONT></P><P><FONT SIZE="2">7. GENERAL</FONT></P><P><FONT SIZE="2">If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.</FONT></P><P><FONT SIZE="2"></FONT></P><P><FONT SIZE="2">If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.</FONT></P><P><FONT SIZE="2"></FONT></P><P><FONT SIZE="2">All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.</FONT></P><P><FONT SIZE="2"></FONT></P><P><FONT SIZE="2">Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may
assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistributions and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2000-2003, jMock.org
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of jMock nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html;charset=utf-8" />
<title>EasyMock License</title>
<link rel="stylesheet" href="easymock.css" />
</head>

<body><div class="bodywidth">

<h2>
EasyMock 2 License (MIT License)

Copyright (c) 2001-2006 OFFIS, Tammo Freese.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
oustanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.216 okio 2.7.0
1.216.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/-Util.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/internal/-Utf8.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/-Platform.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-Platform.kt

No license file was found, but licenses were detected in source scan.

*/
* Copyright 2014 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ByteString.kt

No license file was found, but licenses were detected in source scan.

*/
* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ForwardingTimeout.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/SegmentPool.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Segment.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Sink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Buffer.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Segment.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2014 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/SegmentPool.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Segment.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Sink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Buffer.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Segment.kt

Open Source Used In IOx-FogDirector-AC18 1.17.0 1919
jar/jvmMain/okio/AsyncTimeout.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/JvmOkio.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/SegmentPool.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/GzipSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/DeflaterSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ForwardingSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Source.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/RealBufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Utf8.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/InflaterSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/BufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/ForwardingSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/BufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Timeout.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/GzipSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Utf8.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s): 
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2018 Square, Inc.
 */
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
* /

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Options.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/HashTable.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/PeekSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Options.kt

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (C) 2016 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *    http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Pipe.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Pipe.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Pipe.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Options.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Options.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/jvmMain/okio/Options.kt
Jar file contents:

```
/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2.7.0-sources-jar/commonMain/okio/internal/RealBufferedSink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2.7.0-sources-jar/commonMain/okio/internal/RealBufferedSource.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2.7.0-sources-jar/commonMain/okio/internal/SegmentedByteString.kt

No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2.7.0-sources-jar/commonMain/okio/Sink.kt
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2.7.0-sources-jar/commonMain/okio/BufferedSink.kt

No license file was found, but licenses were detected in source scan.
```
No license file was found, but licenses were detected in source scan.

/*
 * Licensed to the Apache Software Foundation (ASF) under one or more
 * contributor license agreements. See the NOTICE file distributed with
 * this work for additional information regarding copyright ownership.
 * The ASF licenses this file to You under the Apache License, Version 2.0
 * (the "License"); you may not use this file except in compliance with
 * the License. You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1074323481_1595745315.39/0/okio-2-7-0-sources-jar/commonMain/okio/Okio.kt

1.217 shiro-event 1.7.1
1.217.1 Available under license:
Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
The implementation for org.apache.shiro.util.SoftHashMap is based on initial ideas from Dr. Heinz Kabutz's publicly posted version available at http://www.javaspecialists.eu/archive/Issue015.html, with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.218 jackson-datatype-jsr310 2.11.2
1.218.1 Available under license :
This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.219 zlib 1.2.11.dfsg 0ubuntu2
1.219.1 Available under license :
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.220 metrics-health-checks 4.1.9

1.220.1 Available under license:
Apache-2.0

1.221 cracklib 2.9.2-5build1

1.221.1 Available under license:
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: cracklib
Upstream-Contact: Nathan Neulinger <nneul@umr.edu>
Source: http://sourceforge.net/projects/cracklib
Files: *
Copyright: 1993 Alec Muffett <alecm@crypto.dircon.co.uk>
2005-2014 Nathan Neulinger <nneul@umr.edu>
2008-2014 Jan Dittberner <jan@dittberner.info>
License: LGPL-2.1
Comment:
Starting from version 2.8.15 released on 2009-11-19 cracklib is licensed under the terms of the LGPL 2.1. For a detailed discussion and history see README-LICENSE.

Files: debian/*
Copyright: 1998 Jean Pierre LeJacq
Martin Pitt <martin@piware.de>
Jan Dittberner <jandd@debian.org>
License: LGPL-2.1

License: LGPL-2.1
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

A copy of the GNU Lesser General Public License 2.1 is available as
/usr/share/common-licenses/LGPL-2.1 in the Debian GNU/Linux
distribution or on the World Wide Web at
http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html. You can also
obtain it by writing to the Free Software Foundation, Inc., 51
Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

1.222 jackson 2.10.0

1.222.1 Available under license :

========================================================================
== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0, ==
== in this case for the Apache Camel distribution. ==
========================================================================

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Please read the different LICENSE files present in the licenses directory of
this distribution.

Camel :: Jackson
Copyright 2007-2012 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.223 libheimntlmheimdal 7.5.0+dfsg-1

1.223.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Please see info documentation for the complete list of licenses.

Copyright (c) 1992, 1993

The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
              lib/wind/rfc3490.txt
              lib/wind/rfc3491.txt
              lib/wind/rfc4013.txt
              lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
          2009, 2010 Apple, Inc.
          1991, 2013 Massachusetts Institute of Technology
          1995-1997 Eric Young <eay@mincom.oz.au>
          1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software.
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
On Debian systems, the full text can be found in
/usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#.
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
#.
# 1. The authors are not responsible for the consequences of use of this
#    software, no matter how awful, even if they arise from flaws in it.
#.
# 2. The origin of this software must not be misrepresented, either by
#    explicit claim or by omission. Since few users ever read sources,
#    credits must appear in the documentation.
#.
# 3. Altered versions must be plainly marked as such, and must not be
#    misrepresented as being the original software. Since few users
#    ever read sources, credits must appear in the documentation.
#.
# 4. This notice may not be removed or altered.

@macro copynext{}
Portions Copyright (c) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.
WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

@end verbatim
@copynext

@heading The Regents of the University of California

The parts of the libroken, most of libtelnet, telnet, ftp, and popper.

@verbatim
Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS `AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading The Regents of the University of California.

libedit

@verbatim
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath
LibTomMath is hereby released into the Public Domain.

Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Marko Kreen

Copyright (c) 2005 Marko Kreen
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@end verbatim

Copyright (c) 2006, 2007
NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@verbatim
@end verbatim
@verbatim

@verbatim
Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS `AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@verbatim

@verbatim

AES in libhcrypto

@end verbatim
@verbatim

@verbatim
rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim

Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior
permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@copynext

@heading Richard Outerbridge

DES core in libcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec's THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Open Source Used In IOx-FogDirector-AC18 1.17.0 1947
- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext
@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Novell nor the names of its contributors may be used
to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copyrightend

1.224 shiro-aspectj 1.7.1
1.224.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based on initial ideas from Dr. Heinz Kabutz's publicly posted version available at http://www.javaspecialists.eu/archive/Issue015.html, with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[ ]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.225 spring-plugin-core 1.2.0.RELEASE

1.225.1 Available under license :

Apache License, Version 2.0
http://www.apache.org/licenses/LICENSE-2.0
Copyright 2010 the original author or authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
See the License for the specific language governing permissions and
limitations under the License.

1.226 legacy8ujsse 1.1.1

1.226.1 Available under license :

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license
document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share
and change it. By contrast, the GNU General Public License is intended to
guarantee your freedom to share and change free software--to make sure the
software is free for all its users. This General Public License applies to
most of the Free Software Foundation's software and to any other program whose
authors commit to using it. (Some other Free Software Foundation software is
covered by the GNU Library General Public License instead.) You can apply it to
your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your
rights to work written entirely by you; rather, the intent is to exercise the
right to control the distribution of derivative or collective works based on
the Program.

In addition, mere aggregation of another work not based on the Program with the
Program (or with a work based on the Program) on a volume of a storage or
distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under
Section 2) in object code or executable form under the terms of Sections 1 and
2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source
code, which must be distributed under the terms of Sections 1 and 2 above
on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to
give any third party, for a charge no more than your cost of physically
performing source distribution, a complete machine-readable copy of the
corresponding source code, to be distributed under the terms of Sections 1
and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed only
for noncommercial distribution and only if you received the program in
object code or executable form with such an offer, in accord with
Subsection b above.)

The source code for a work means the preferred form of the work for making
modifications to it. For an executable work, complete source code means all
the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation
of the executable. However, as a special exception, the source code
distributed need not include anything that is normally distributed (in either
source or binary form) with the major components (compiler, kernel, and so on)
of the operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the source
code from the same place counts as distribution of the source code, even though
third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as
expressly provided under this License. Any attempt otherwise to copy, modify,
sublicense or distribute the Program is void, and will automatically terminate
your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school,
if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.227 hikaricp-java7 2.4.13
1.228 slf4j 1.7.30

1.228.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.229 javassist 3.19.0-GA

1.229.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
 *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 */

Found in path(s):
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/proxy/RuntimeSupport.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-
jar/javassist/tools/rmi/ObjectNotFoundException.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/ast/CastExpr.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/convert/TransformBefore.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/convert/TransformFieldAccess.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/analysis/IntQueue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/tools/rmi/AppletServer.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/HotSwapper.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/tools/reflect/Metalevel.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/AnnotationImpl.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/proxy/MethodFilter.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/ast/DoubleConst.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/TypeChecker.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/CtMember.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/JvstCodeGen.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/ClassClassPath.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/Opcode.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/tools/reflect/Sample.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/MemberCodeGen.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/ctf/PrimitiveType.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/URLClassPath.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/SignatureAttribute.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/CodeIterator.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/stackmap/Tracer.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/proxy/MethodHandler.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/proxy/ProxyObject.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/LoaderClassPath.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/ast/MethodDecl.java
Open Source Used In IOx-FogDirector-AC18 1.17.0

* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/ByteArrayClassPath.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/util/proxy/SecurityActions.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/annotation/AnnotationsWriter.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/BinExpr.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/FieldDecl.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/analysis/Type.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/runtime/Cflow.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/Mnemonic.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/SymbolTable.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/convert/TransformCall.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/AssignExpr.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/EnclosingMethodAttribute.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/CtNewNestedClass.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/exp/FieldAccess.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/util/proxy/Proxy.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/AssignExpr.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/InnerClassesAttribute.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/tools/Dump.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/stackmap/TypedBlock.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/analysis/Subroutine.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/Lex.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/bytecode/analysis/Subroutine.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/Lex.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/Variable.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/tools/LoadingAttribute.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1.jar/javassist/compiler/ast/FieldAccess.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 1964
Open Source Used In IOx-FogDirector-AC18 1.17.0  1966
No license file was found, but licenses were detected in source scan.

/*
 * Javassist, a Java-bytecode translator toolkit.
 * Copyright (C) 2004 Bill Burke. All Rights Reserved.
 *
 * The contents of this file are subject to the Mozilla Public License Version
 * 1.1 (the "License"); you may not use this file except in compliance with
 * the License. Alternatively, the contents of this file may be used under
 * the terms of the GNU Lesser General Public License Version 2.1 or later,
 * or the Apache License Version 2.0.
 *
 * Software distributed under the License is distributed on an "AS IS" basis,
 * WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License
 * for the specific language governing rights and limitations under the
 * License.
 */

Found in path(s):

* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/ast/CondExpr.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/util/proxy/ProxyObjectInputStream.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/compiler/Javac.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/ByteMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/ClassMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/ShortMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/AnnotationMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/MemberValueVisitor.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/StringMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/Annotation.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/EnumMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/ArrayMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/DoubleMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/CharMemberValue.java
* /opt/cola/permits/1135902017_1613627678.46/0/javassist-3-19-0-ga-sources-1-jar/javassist/bytecode/annotation/BooleanMemberValue.java
1.230 commons-codec 1.5

1.230.1 Available under license:

Apache Commons Codec
Copyright 2002-2011 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains
test data from http://aspell.sourceforge.net/test/batch0.tab.

Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org). Verbatim copying
and distribution of this entire article is permitted in any medium,
provided this notice is preserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, 
publicly display, publicly perform, sublicense, and distribute the 
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of 
this License, each Contributor hereby grants to You a perpetual, 
worldwide, non-exclusive, no-charge, royalty-free, irrevocable 
(except as stated in this section) patent license to make, have made, 
use, offer to sell, sell, import, and otherwise transfer the Work, 
where such license applies only to those patent claims licensable 
by such Contributor that are necessarily infringed by their 
Contribution(s) alone or by combination of their Contribution(s) 
with the Work to which such Contribution(s) was submitted. If You 
institute patent litigation against any entity (including a 
cross-claim or counterclaim in a lawsuit) alleging that the Work 
or a Contribution incorporated within the Work constitutes direct 
or contributory patent infringement, then any patent licenses 
granted to You under this License for that Work shall terminate 
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the 
Work or Derivative Works thereof in any medium, with or without 
modifications, and in Source or Object form, provided that You 
meet the following conditions:

(a) You must give any other recipients of the Work or 
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices 
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works 
that You distribute, all copyright, patent, trademark, and 
attribution notices from the Source form of the Work, 
excluding those notices that do not pertain to any part of 
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its 
distribution, then any Derivative Works that You distribute must 
include a readable copy of the attribution notices contained 
within such NOTICE file, excluding those notices that do not 
pertain to any part of the Derivative Works, in at least one 
of the following places: within a NOTICE text file distributed 
as part of the Derivative Works; within the Source form or 
documentation, if provided along with the Derivative Works; or, 
within a display generated by the Derivative Works, if and 
wherever such third-party notices normally appear. The contents 
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.231 hk2-utils 2.6.1

1.231.1 Available under license :

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code,
"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's
receipt of the Program under the terms of a Secondary License
(if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   
   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement,
including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate
entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is
intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is implemented
by public license practices. Many people have made generous
contributions to the wide range of software distributed through that
system in reliance on consistent application of that system; it is up to
the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be
a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may
add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new
versions of the General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY
11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR
DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL
DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM
(INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF
THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR
OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively convey
the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Public License v. 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made
available under the following Secondary Licenses when the conditions for such
availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU
General Public License, version 2 with the GNU Classpath Exception which is

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish
## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.232 nghttp2 1.30.0-1ubuntu1

1.232.1 Available under license :

The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```plaintext
set yrange [0:]
set terminal pngcairo font 'Sans, 8' lw 1 size 1400,1024
set xtics rotate by -45
set style histogram errorbars gap 2 lw 1
set style fill solid border -1
```
LEGAL NOTICE INFORMATION

All the files in this distribution are covered under the MIT license (see the file MITL) except some files mentioned below:
If not otherwise noted, the extensions in this package are licensed under the following license.

Copyright (c) 2010 by the contributors (see AUTHORS file).
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.233 docker-java-transport-httpclient 3.2.5
1.233.1 Available under license :

Found license 'GNU Lesser General Public License' in *

* This library is free software; you can redistribute it and/or modify it under *
  the terms of the GNU Lesser General Public License as published by the Free *
  Software Foundation; either version 2.1 of the License, or (at your option) *
  any later version. * This library is distributed in the hope that it will be useful, but *
  WITHOUT ANY WARRANTY; without even the implied warranty of *
  MERCHANTABILITY or * FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public *
  License for more details. * You should have received a copy of the GNU Lesser General Public License along *
  Redistribution and use in source and binary forms, with or without modification, * are permitted provided that the *
  following conditions are met: * Redistributions of source code must retain the above copyright *
  notice, this list * of conditions and the following disclaimer. * Redistributions in binary form must reproduce the above copyright *
  notice, this * list of conditions and the following disclaimer in the documentation and/or * Neither the name of the
1.234 yamlbeans 1.09

1.235 istack-commons 3.0.7

1.235.1 Available under license:

Found license 'General Public License 2.0' in 'Copyright (c) 1997-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 1997-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

1.236 kotlin 1.3.70

1.236.1 Available under license:

No license file was found, but licenses were detected in source scan.

{"version":3,"file":"kotlin.js","sources":[]}
ArrayBuffer.isView(a)\n\nKotlin.arrayToString = function (a) {
    var toString = Kotlin.isCharArray(a) ? String.fromCharCode : Kotlin.toString;
    return "[" + Array.prototype.map.call(a, function(e) { return toString(e); }).join(", ") + "]";
}

Kotlin.arrayDeepToString = function (arr) {
    return Kotlin.kotlin.collections.contentDeepToStringImpl(arr);
}

Kotlin.arrayEquals = function (a, b) {
    if (a === b) { return true; }
    if (!Kotlin.isArrayish(b)) { return false; }
    var i = 0, n = a.length; for (var i = 0, n = a.length; i < n; i++) { return false; }
}

Kotlin.arrayHashCode = function (arr) {
    var result = 1;
    for (var i = 0, n = arr.length; i < n; i++) { result = ((31 * result | 0) + Kotlin.hashCode(arr[i]));
} return result;
}

Kotlin.propertyRefClassMetadataCache = [
    {
        mutable: { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutableProperty0 }
        },
        immutable: { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KProperty0 }
    },
    {
        mutable: { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KMutableProperty1 }
        },
        immutable: { value: null, implementedInterface: function () {
            return Kotlin.kotlin.reflect.KProperty1 }
    }
]

Kotlin.equals = function (obj1, obj2) {
    if (obj1 == null) {
        return obj2 == null;
    }
    if (obj2 == null) {
        return false;
    }
    if (typeof obj1 === "object" && typeof obj1.equals === "function") {
        return obj1.equals(obj2);
    }
    if (typeof obj1 === "number" && typeof obj2 === "number") {
        return obj1 === obj2 && (obj1 !== 0 || 1 / obj1 === 1 / obj2)
    }
    return obj1 === obj2;
}

Kotlin.hashCode = function (obj) {
    if (obj == null) {
        return 0;
    }
    var objType = typeof obj;
    switch (objType) {
        case "function":
            return Kotlin.kotlin.reflect.KFunction0.hashCode(obj);
        case "object":
            return Kotlin.kotlin.reflect.KObject0.hashCode(obj);
        case "array":
            return Kotlin.kotlin.reflect.KArray0.hashCode(obj);
        case "boolean":
            return obj ? 0 : 1;
        case "number":
            return obj | 0;
        case "null":
            return 0;
        case "string":
            return Kotlin.kotlin.reflect.KString0.hashCode(obj);
        case "undefined":
            return 0;
        default:
            return obj || 0;
    }
}
== typeof obj.hashCode ? obj.hashCode() : getObjectHashCode(obj);\n    }\n    if ("function" === objType) {\n        return getObjectHashCode(obj);\n    }\n    if ("object" === objType) {\n        return Number(obj);\n    }\n    if ("number" === objType) {
        return Kotlin.numberHashCode(obj);
    }\n    if ("boolean" === objType) {\n        return Number(obj);
    }\n    if ("null" === obj) {\n        return null;\n    }\n    var str = String(obj);\n    return getHexStringCode(str);\n}\n\nfunction getHexStringCode(str) {\n    var hash = 0;\n    for (var i = 0; i < str.length; i++) {\n        var code = str.charCodeAt(i);\n        hash = (hash * 16 + code) | 0; // Keep high 32-bit.\n    }\n    return hash;
}
```javascript
Kotlin.Long.TWO_PWR_63_DBL_ = 0

Kotlin.Long.TWO_PWR_32_DBL_ = 0

Kotlin.Long.TWO_PWR_31_DBL_ = 0

Kotlin.Long.TWO_PWR_48_DBL_ = 0

Kotlin.Long.TWO_PWR_64_DBL_ = 0

Kotlin.Long.ZERO = Kotlin.Long.MAX_VALUE

Kotlin.Long.ONE = Kotlin.Long.MIN_VALUE

Kotlin.Long.NEG_ONE = Kotlin.Long.MAX_VALUE

Kotlin.Long.prototype.toNumber = function() {
  return this.high_ * Kotlin.Long.TWO_PWR_32_DBL_ +
  this.getLowBitsUnsigned();
}
```

// We need to
change the Long value before it can be negated, so we remove
   // the bottom-most digit in this base and then
   recur to do the rest.
   var radixLong = Kotlin.Long.fromNumber(radix);
   var div = this.div(radixLong);
   var rem = div.multiply(radixLong).subtract(this);
   return div.toString(radix) + rem.toInt().toString(radix);
}

else {
   return '-' + this.negate().toString(radix);
}

// Do several (6) digits each time through the loop,
// so as to minimize the calls to the very expensive emulated div.
var radixToPower = Kotlin.Long.fromNumber(Math.pow(radix, 6));

var rem = this;
var result = '';
while (true) {
   var remDiv = rem.div(radixToPower);
   var intval = rem.subtract(remDiv.multiply(radixToPower)).toInt();
   var digits = intval.toString(radix);

   rem = remDiv;
   if (rem.isZero()) {
      return digits + result;
   } else {
      while (digits.length < 6) {
         digits = '0' + digits;
      }
      result = '' + digits + result;
   }
}

/**
 * @return {number} The high 32-bits as a signed value.
 */
Kotlin.Long.prototype.getHighBits = function() {
   return this.high_;
};

/** @return {number} The low 32-bits as a signed value.
 */
Kotlin.Long.prototype.getLowBits = function() {
   return this.low_;
};

/** @return {number} The low 32-bits as an unsigned value.
 */
Kotlin.Long.prototype.getLowBitsUnsigned = function() {
   return (this.low_ >= 0) ?
      this.low_ : Kotlin.Long.TWO_PWR_32_DBL_ + this.low_;
};

/**
 * @return {number} Returns the number of bits needed to represent the absolute
 *     value of this Long.
 */
Kotlin.Long.prototype.getNumBitsAbs = function() {
   if (this.isNegative()) {
      if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
         return 64;
      } else {
         return this.negate().getNumBitsAbs();
      }
   } else {
      var val = this.high_ != 0 ? this.high_ : this.low_;
      for (var bit = 31; bit > 0; bit--) {
         if ((val & (1 << bit)) != 0) {
            break;
         }
      }
      return this.high_ != 0 ? bit + 33
   } else {
      return bit + 1;
   }
};

/** @return {boolean} Whether this value is zero.
 */
Kotlin.Long.prototype.isZero = function() {
   return this.high_ == 0 && this.low_ == 0;
};

/** @return {boolean} Whether this value is negative.
 */
Kotlin.Long.prototype.isNegative = function() {
   return this.high_ < 0;
};

/** @return {boolean} Whether this value is odd.
 */
Kotlin.Long.prototype.isOdd = function() {
   return (this.low_ & 1) == 1;
};

/** @return {boolean} Whether this value is less than the other.
 */
Kotlin.Long.prototype.lessThan = function(other) {
   return this.compare(other) < 0;
};

/** @return {boolean} Whether this value is less than or equal to the other.
 */
Kotlin.Long.prototype.lessThanOrEqual = function(other) {
   return this.compare(other) <= 0;
};

/** @return {boolean} Whether this value is greater than the other.
 */
Kotlin.Long.prototype.greaterThan = function(other) {
   return this.compare(other) > 0;
};

/** @return {boolean} Whether this value is greater than or equal to the other.
 */
Kotlin.Long.prototype.greaterThanOrEqual = function(other) {
   return this.compare(other) >= 0;
};

/** @return {boolean} The negation of this value.
 */
Kotlin.Long.prototype.negate = function() {
   if (this.equalsLong(Kotlin.Long.MIN_VALUE)) {
      return Kotlin.Long.MIN_VALUE;
   } else {
      return this.not().add(Kotlin.Long.ONE);
   }
};

/** @param {Kotlin.Long} other Long to add to this one.
 */
Kotlin.Long.prototype.add = function(other) {
   return this.subtract(other).add(Kotlin.Long.ONE);
};
then sum the chunks.

```
var a48 = this.high_ >>> 16;
var a32 = this.high_ & 0xFFFF;
var a16 = this.high_ & 0xFFFF;
var a00 = this.low_ >>> 16;
var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
var b48 = other.high_ & 0xFFFF;
var b32 = other.high_ & 0xFFFF;
var b16 = other.low_ & 0xFFFF;
var b00 = other.low_ & 0xFFFF;
var c48 = 0, c32 = 0, c16 = 0, c00 = 0;
var c48 += a48 + b48;
var c32 += a32 + b32;
var c16 += a16 + b16;
var c00 += a00 + b00;
var c48 &= 0xFFFF;
var c32 &= 0xFFFF;
var c16 &= 0xFFFF;
var c00 &= 0xFFFF;
```
We will tweak the approximate result by changing it in the 48-th digit or

// the smallest non-fractional digit, whichever is larger.

var log2 = Math.ceil(Math.log(approx) / Math.LN2);

var delta = (log2 <= 48) ? 1 :

Math.pow(2, log2 - 48);

// Decrease the approximation until it is smaller than the remainder. Note

// that if it is too large, the product overflows and is negative.

var approxRes = Kotlin.Long.fromNumber(approx);

var approxRem = approxRes.multiply(other);

while (approxRem.isNegative() || approxRem.greaterThan(rem)) {

  approx -= delta;

  approxRes = Kotlin.Long.fromNumber(approx);

  approxRem = approxRes.multiply(other);
}

// We know the answer can't be zero... and actually, zero would cause

// infinite recursion since we would make no progress.

if (approxRes.isZero()) {

  approxRes =

  Kotlin.Long.ONE;
}

res = res.add(approxRes);

rem = rem.subtract(approxRem);

}

return res;
};

/**
 * Returns this Long modulo the given one.
 * @param {Kotlin.Long} other Long by which to mod.
 * @return {Kotlin.Long} This Long modulo the given one.
 */
Kotlin.Long.prototype.modulo =

function(other) {
  return this.subtract(this.div(other).multiply(other));
};

/** @return {Kotlin.Long} The

bitwise-NOT of this value.
 */
Kotlin.Long.prototype.not =

function() {
  return Kotlin.Long.fromBits(~this.low_,

  ~this.high_);
};

/**
 * Returns the bitwise-AND of this Long and the given one.
 * @param {Kotlin.Long} other Long with which to AND.
 * @return {Kotlin.Long} The bitwise-AND of this and the other.
 */
Kotlin.Long.prototype.and =

function(other) {
  return Kotlin.Long.fromBits(this.low_ & other.low_,

  this.high_ & other.high_);
};

/**
 * Returns the bitwise-XOR of this Long and the given one.
 * @param {Kotlin.Long} other Long with which to XOR.
 * @return {Kotlin.Long} The bitwise-XOR of this and the other.
 */
Kotlin.Long.prototype.xor =

function(other) {
  return Kotlin.Long.fromBits(this.low_ ^ other.low_,

  this.high_ ^ other.high_);
};

/**
 * Returns this Long with bits shifted to the left by the given amount.
 * @param {number} numBits The number of bits by which to shift.
 * @return {Kotlin.Long} This shifted to the left by the given amount.
 */
Kotlin.Long.prototype.shiftLeft =

function(numBits) {
  numBits &= 63;
  if (numBits == 0) {
    return this;
  } else {
    var high = this.high_;
    if (numBits < 32) {
      var low = this.low_;
      return Kotlin.Long.fromBits(

          low << numBits,

          (high << numBits) | (low >>> (32 - numBits)));
    } else if (numBits == 32) {
      return Kotlin.Long.fromBits(high, 0);
    } else {
      return Kotlin.Long.fromBits(high >> (numBits - 32),

          high >= 0 ? 0 : -1);
    }
  }
}

// Support for

Kotlin

Kotlin.Long.prototype.equals = function (other) {
    return other instanceof Kotlin.Long &&
    this.equalsLong(other);
};

Kotlin.Long.prototype.compareTo_11rb$ =

Kotlin.Long.prototype.compare;

Kotlin.Long.prototype.increment =

function() {
    return this.add(Kotlin.Long.ONE);
};

Kotlin.Long.prototype.decrement =

function() {
    return this.add(Kotlin.Long.NEG_ONE);
};

Kotlin.Long.prototype.valueOf =

function() {
    return

this.toNumber();
};

Kotlin.Long.prototype.unaryPlus =

function() {
    return

```javascript
Math.atanh = function(x) {
            if (Math.abs(x) < taylor_n_bound) {
                var result = x;
            }
            return Math.sqrt(2) * result;
        }

        if (typeof Math.atanh === "undefined") {
            Math.acosh = function(x) {
                if (x < 1) {
                    return NaN;
                }
                if (x > 0) {
                    return Math.floor(x);
                }
                return Math.ceil(x);
            }
        }

        Math.asinh = function(x) {
            if (Math.abs(x) >= taylor_2_bound) {
                var y = Math.sqrt(x - 1);
                // approximation by taylor series in y at 0 up to order 4
                result -= y3 / 12;
            }
            return Math.log(x + Math.sqrt(x * x - 1));
        }

        if (typeof Math.asinh === "undefined") {
            Math.tanh = function(x) {
                if (x <= -taylor_n_bound) {
                    return Math.exp(+x) / (Math.exp(+x) + Math.exp(-x));
                }
                if (x >= +taylor_n_bound) {
                    return Math.exp(-x) / (Math.exp(+x) + Math.exp(-x));
                }
                return result -= (x * x * x) / 3;
            }
        }

        Math.cosh = function(x) {
            var y = Math.exp(x);
            var y1 = 1 / y;
            if (!isFinite(y)) return Math.exp(x - Math.LN2);
            if (isFinite(y)) return Math.exp(x - Math.LN2);
            if (isFinite(y1)) return (y - y1) / 2;
        }

        Math.sinh = function(x) {
            if (Math.abs(x) < taylor_n_bound) {
                var result = x;
            } else {
                var y = Math.exp(x);
                var y1 = 1 / y;
                if (!isFinite(y) || !isFinite(y1)) return Math.exp(Math.abs(x) - Math.LN2);
                if (isFinite(y)) return Math.exp(x - Math.LN2);
                if (isFinite(y1)) return -Math.exp(-x - Math.LN2);
            }
            return Math.LN2;
        }

        Math.tanh = function(x) {
            if (x <= -taylor_n_bound) {
                return Math.exp(+x) / (Math.exp(+x) + Math.exp(-x));
            }
            if (x >= +taylor_n_bound) {
                return Math.exp(-x) / (Math.exp(+x) + Math.exp(-x));
            }
            return result -= (x * x * x) / 3;
        }
```

The code above implements the inverse hyperbolic functions (asinh, atanh, acosh, acosh, sinh, and cosh) using Taylor series approximations. It includes handling for special cases like NaN, 0, and infinity to ensure numerical stability.

The code also includes a module for `Math.sign`, `Math.trunc`, and `Math.trunc` which are not native JavaScript functions. These are likely providing additional utility functions for rounding and sign determination.

The comments indicate that the code is part of an open-source project called IOx-FogDirector-AC18 and is distributed under the Apache 2.0 license. The copyright notice is from 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
if (Math.abs(x) > taylor_2_bound) {\n
result += (x * x * x) / 3;\n
}\n
return result;\n
}\n
return Math.log((1 + x) / (1 - x)) / 2;\n
};\n }\n if (typeof Math.log1p === \"undefined\") {\n
Math.log1p = function(x) {\n
if (Math.abs(x) < taylor_n_bound) {\n
var x2 = x * x;\n
var x3 = x2 * x;\n
var x4 = x3 * x;\n
// approximation by taylor series in x at 0 up to order 4\n
return (-x4 / 4 + x3 / 3 - x2 / 2 + x);\n
}\n
return Math.log(x + 1);\n
};\n }\n if (typeof
Math.expm1 === \"undefined\") {\n
Math.expm1 = function(x) {\n
if (Math.abs(x) < taylor_n_bound)
{\n
var x2 = x * x;\n
var x3 = x2 * x;\n
var x4 = x3 * x;\n
// approximation by
taylor series in x at 0 up to order 4\n
return (x4 / 24 + x3 / 6 + x2 / 2 + x);\n
}\n
return
Math.exp(x) - 1;\n
};\n }\n})();\nif (typeof Math.hypot === \"undefined\") {\n Math.hypot = function() {\n
var y = 0;\n
var length = arguments.length;\n\n
for (var i = 0; i < length; i++) {\n
if (arguments[i]
=== Infinity || arguments[i] === -Infinity) {\n
return Infinity;\n
}\n
y += arguments[i] *
arguments[i];\n
}\n
return Math.sqrt(y);\n };\n}\nif (typeof Math.log10 === \"undefined\") {\n
Math.log10 = function(x) {\n
return Math.log(x) * Math.LOG10E;\n };\n}\nif (typeof Math.log2 ===
\"undefined\") {\n Math.log2 = function(x) {\n
return Math.log(x) * Math.LOG2E;\n };\n}\nif (typeof
Math.clz32 === \"undefined\") {\n Math.clz32 = (function(log, LN2) {\n
return function(x) {\n
var
asUint = x >>> 0;\n
if (asUint === 0) {\n
return 32;\n
}\n
return 31 - (log(asUint) /
LN2 | 0) | 0; // the \"| 0\" acts like math.floor\n
};\n })(Math.log, Math.LN2);\n}\n\n// For HtmlUnit and
PhantomJs\nif (typeof ArrayBuffer.isView === \"undefined\") {\n ArrayBuffer.isView = function(a) {\n
return a != null && a.__proto__ != null && a.__proto__.__proto__ === Int8Array.prototype.__proto__;\n
};\n}\n\nif (typeof Array.prototype.fill === \"undefined\") {\n // Polyfill from https://developer.mozilla.org/enUS/docs/Web/JavaScript/Reference/Global_Objects/Array/fill#Polyfill\n Array.prototype.fill = function() {\n
//
Steps 1-2.\n
if (this == null) {\n
throw new TypeError('this is null or not defined');\n
}\n\n
var O =
Object(this);\n\n
// Steps 3-5.\n
var len = O.length >>> 0;\n\n
// Steps 6-7.\n
var start = arguments[1];\n
var relativeStart = start >> 0;\n\n
// Step 8.\n
var k = relativeStart < 0 ?\n
Math.max(len + relativeStart,
0) :\n
Math.min(relativeStart, len);\n\n
// Steps 9-10.\n
var end = arguments[2];\n
var relativeEnd =
end === undefined ?\n
len : end >> 0;\n\n
// Step 11.\n
var final = relativeEnd < 0 ?\n
Math.max(len
+ relativeEnd, 0) :\n
Math.min(relativeEnd, len);\n\n
// Step 12.\n
while (k < final) {\n
O[k] =
value;\n
k++;\n
}\n\n
// Step 13.\n
return O;\n };\n}\n\n(function() {\n function
normalizeOffset(offset, length) {\n
if (offset < 0) return Math.max(0, offset + length);\n
return
Math.min(offset, length);\n }\n function typedArraySlice(begin, end) {\n
if (typeof end === \"undefined\")
{\n
end = this.length;\n
}\n
begin = normalizeOffset(begin || 0, this.length);\n
end =
Math.max(begin, normalizeOffset(end, this.length));\n
return new this.constructor(this.subarray(begin, end));\n
}\n\n var arrays = [Int8Array, Int16Array, Uint16Array, Int32Array, Float32Array, Float64Array];\n for (var i =
0; i < arrays.length; ++i) {\n
var TypedArray = arrays[i];\n
if (typeof TypedArray.prototype.fill ===
\"undefined\") {\n
TypedArray.prototype.fill = Array.prototype.fill;\n
}\n
if (typeof
TypedArray.prototype.slice === \"undefined\") {\n
Object.defineProperty(TypedArray.prototype, 'slice', {\n
value: typedArraySlice\n
});\n
}\n }\n\n // Patch apply to work with TypedArrays if needed.\n
try {\n
(function() {}).apply(null, new Int32Array(0))\n } catch (e) {\n
var apply =
Function.prototype.apply;\n
Object.defineProperty(Function.prototype, 'apply', {\n
value: function(self,
array) {\n
return apply.call(this, self, [].slice.call(array));\n
}\n
});\n }\n\n\n // Patch map to
work with TypedArrays if needed.\n for (var i = 0; i < arrays.length; ++i) {\n
var TypedArray = arrays[i];\n
if (typeof TypedArray.prototype.map === \"undefined\") {\n
Object.defineProperty(TypedArray.prototype,
'map', {\n
value: function(callback, self) {\n
return [].slice.call(this).map(callback, self);\n
}\n
});\n
}\n }\n\n // Patch sort to work with TypedArrays if needed.\n // TODO: consider to
remove following function and replace it with `Kotlin.doubleCompareTo` (see misc.js)\n var
totalOrderComparator = function (a, b) {\n
if (a < b) return -1;\n
if (a > b) return 1;\n\n
if (a === b) {\n
if (a !== 0) return 0;\n\n
var ia = 1 / a;\n
return ia === 1 / b ? 0 : (ia < 0 ? -1 : 1);\n
}\n\n
return a !== a ? (b !== b ? 0 : 1) : -1\n };\n\n for (var i = 0; i < arrays.length; ++i) {\n
var TypedArray =

Open Source Used In IOx-FogDirector-AC18 1.17.0 2002


arrays[i];
        if (typeof TypedArray.prototype.sort === "undefined") {
            Object.defineProperty(TypedArray.prototype, 'sort', {
                value: function(compareFunction) {
                    return Array.prototype.sort.call(this, compareFunction);
                }
            });
        }
    }

})(function() {
/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

Kotlin.Kind = {
    CLASS: "class",
    INTERFACE: "interface",
    OBJECT:
//

Kotlin.callGetter = function (thisObject, klass, propertyName) {
    var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);
    if (propertyDescriptor != null && propertyDescriptor.get != null) {
        return propertyDescriptor.get.call(thisObject);
    }

    propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject, propertyName);
    if (propertyDescriptor != null && "value" in propertyDescriptor) {
        return thisObject[propertyName];
    }

    return Kotlin.callGetter(thisObject, Object.getPrototypeOf(klass), propertyName);
}

Kotlin.callSetter = function (thisObject,(klass, propertyName, value) {
    var propertyDescriptor = Object.getOwnPropertyDescriptor(klass, propertyName);
    if (propertyDescriptor != null && propertyDescriptor.set != null) {
        propertyDescriptor.set.call(thisObject, value);
        return;
    }

    propertyDescriptor = Object.getOwnPropertyDescriptor(thisObject, propertyName);
    if (propertyDescriptor != null && "value" in propertyDescriptor) {
        thisObject[propertyName] = value;
        return
    }

    Kotlin.callSetter(thisObject, Object.getPrototypeOf(klass), propertyName, value);
}

function isInheritanceFromInterface(ctor, iface) {
    if (ctor === iface) return true;

    var metadata = ctor.$metadata$;
    if (metadata != null) {
        var interfaces = metadata.interfaces;
        for (var i = 0; i < interfaces.length; i++) {
            if (isInheritanceFromInterface(interfaces[i], iface)) {
                return true;
            }
        }
    }

    var superPrototype = ctor.prototype != null ? Object.getPrototypeOf(ctor.prototype) : null;
    var superConstructor = superPrototype.constructor;
    return isInheritanceFromInterface(superConstructor, iface);
}

/**
 * @param {*} object
 * @param {Function|Object} klass
 * @returns {Boolean}
 */
Kotlin.isType = function (object, klass) {
    if (klass === Object) {
        switch (typeof object) {
            case "string":
            case "number":
            case "boolean":
            case "function":
                return true;
            default:
                return object instanceof Object;
        }
    }

    if (object == null || klass == null || (typeof object !== 'object' && typeof object !== 'function')) {
        return false;
    }

    if (typeof klass === 'function' && typeof object !== 'function' && typeof object !== klass) {
        return false;
    }

    var proto = Object.getPrototypeOf(klass);
    var constructor = proto.constructor;
    if (metadata.kind === Kotlin.Kind.OBJECT) {
        return object === klass;
    }

    var classMetadata = klass.$metadata$;
    // In WebKit (JavaScriptCore) for some interfaces from DOM typeof returns "object", nevertheless they can be used in RHS of instanceof
    if (classMetadata == null) {
        return object instanceof klass;
    }

    return klassMetadata.kind === Kotlin.Kind.OBJECT && constructor;}

Kotlin.boxedChar`n var Kotlin.isCharSequence = function (value) {
    return return value instanceof Kotlin.BoxedChar
}

Kotlin.isComparable = function (value) {
    var type = typeof value;
    return type === 'string'||

Kotlin.isType(value, Kotlin.kotlin.Comparable);}

Kotlin.isCharSequence = function (value) {
    return return value instanceof Kotlin.kotlin.CharSequence;
}

/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

package kotlin

public class Enum<T : Enum<T>> :

    val name: String
        get() = _name
    override fun equals(other: Enum<T>) = ordinal.compareTo(other.ordinal)

    override fun hashCode(): Int = js("Kotlin.identityHashCode") (this)


get(1)\n\n/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component2(): Float {\n    return\n    get(1)\n}

/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component2(): Double {\n    return\n    get(1)\n}

/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component2(): Boolean {\n    return\n    get(1)\n}

/**\n * Returns 2nd *element* from the array.\n * \n * If the size of this array is less than 2, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component2(): Char {\n    return\n    get(1)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component3(): T {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component3(): Byte {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component3(): Short {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun IntArray.component3(): Int {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun LongArray.component3(): Long {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun FloatArray.component3(): Float {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun DoubleArray.component3(): Double {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun BooleanArray.component3(): Boolean {\n    return\n    get(2)\n}

/**\n * Returns 3rd *element* from the array.\n * \n * If the size of this array is less than 3, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun CharArray.component3(): Char {\n    return\n    get(2)\n}

/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Array<out T>.component4(): T {\n    return\n    get(3)\n}

/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ByteArray.component4(): Byte {\n    return\n    get(3)\n}

/**\n * Returns 4th *element* from the array.\n * \n * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS\n * where the behavior is unspecified.\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun ShortArray.component4(): Short {\n    return
/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun IntArray.component4(): Int {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun LongArray.component4(): Long {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun FloatArray.component4(): Float {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun DoubleArray.component4(): Double {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun BooleanArray.component4(): Boolean {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * 
 * If the size of this array is less than 4, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun CharArray.component4(): Char {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun <T> Array<out T>.component5(): T {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun ByteArray.component5(): Byte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun ShortArray.component5(): Short {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun IntArray.component5(): Int {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun LongArray.component5(): Long {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun FloatArray.component5(): Float {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun DoubleArray.component5(): Double {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun BooleanArray.component5(): Boolean {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an
 * [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun CharArray.component5(): Char {
    return get(4)
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun
ByteArray.contains(element: Byte): Boolean {\n    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun ShortArray.contains(element: Short): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun IntArray.contains(element: Int): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun LongArray.contains(element: Long): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun FloatArray.contains(element: Float): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun DoubleArray.contains(element: Double): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun BooleanArray.contains(element: Boolean): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns `true` if [element] is found in the array.
 */
public operator fun CharArray.contains(element: Char): Boolean {
    return indexOf(element) >= 0
}

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun <T> Array<out T>.elementAt(index: Int): T

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun ByteArray.elementAt(index: Int): Byte

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun ShortArray.elementAt(index: Int): Short

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun IntArray.elementAt(index: Int): Int

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun LongArray.elementAt(index: Int): Long

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun FloatArray.elementAt(index: Int): Float

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun DoubleArray.elementAt(index: Int): Double

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun BooleanArray.elementAt(index: Int): Boolean

/**
 * Returns an element at the given [index] or throws an
[IndexOutOfBoundsException] if the [index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAt
*/
public expect fun CharArray.elementAt(index: Int): Char

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.elementAtOrElse(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun ByteArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Byte): Byte {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun ShortArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Short): Short {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the
[index] is out of bounds of this array.

@sample samples.collections.Collections.Elements.elementAtOrElse
*/
@kotlin.internal.InlineOnly
public inline fun CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}
IllegalArgumentException: The index is out of bounds for this array.

public inline fun IntArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Int): Int {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Long): Long {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Float): Float {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Double): Double {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or the result of calling the [defaultValue] function if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.elementAtOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.elementAtOrNull(index: Int): Byte? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.elementAtOrNull(index: Int): Short? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.elementAtOrNull(index: Int): Int? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.elementAtOrNull(index: Int): Long? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.elementAtOrNull(index: Int): Float? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.elementAtOrNull(index: Int): Double? {
    return this.getOrNull(index)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 */
@kotlin.internal.InlineOnly
public inline fun BooleanArray.elementAtOrNull(index: Int): Boolean? {
    return this.getOrNull(index)
}
samples.collections.Collections.Elements.elementAtOrNull */
@kotlin.internal.InlineOnly
public inline fun CharArray.elementAtOrNull(index: Int): Char? {
    return this.getOrNull(index)
}

/**
 * Returns the first
 * element matching the given [predicate], or `null` if no such element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.find(predicate: (T) -> Boolean): T? {
    return firstOrNull(predicate)
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.find(predicate: (Byte) -> Boolean): Byte? {
    return firstOrNull(predicate)
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.find(predicate: (Short) -> Boolean): Short? {
    return firstOrNull(predicate)
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.find(predicate: (Int) -> Boolean): Int? {
    return firstOrNull(predicate)
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.find(predicate: (Long) -> Boolean): Long? {
    return firstOrNull(predicate)
}

/**
 * Returns the first
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.find(predicate: (Float) -> Boolean): Float? {
    return firstOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.findLast(predicate: (T) -> Boolean): T? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ByteArray.findLast(predicate: (Byte) -> Boolean): Byte? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun ShortArray.findLast(predicate: (Short) -> Boolean): Short? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun IntArray.findLast(predicate: (Int) -> Boolean): Int? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun LongArray.findLast(predicate: (Long) -> Boolean): Long? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun FloatArray.findLast(predicate: (Float) -> Boolean): Float? {
    return lastOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun DoubleArray.find(predicate: (Double) -> Boolean): Double? {
    return firstOrNull(predicate)
}

/**
 * Returns the last
element matching the given [predicate], or `null` if no such
element was found.
 */
@kotlin.internal.InlineOnly
public inline fun CharArray.findLast(predicate: (Char) -> Boolean): Char? {
    return lastOrNull(predicate)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun <T> Array<out T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ByteArray.first(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
empty.

```java
/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ShortArray.first(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun IntArray.first(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun LongArray.first(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun FloatArray.first(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun DoubleArray.first(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun BooleanArray.first(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun CharArray.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[0]
}
```

```java
/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun <T> Array<out T>.first(predicate: (T) -> Boolean): T {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ByteArray.first(predicate: (Byte) -> Boolean): Byte {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun ShortArray.first(predicate: (Short) -> Boolean): Short {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun IntArray.first(predicate: (Int) -> Boolean): Int {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun LongArray.first(predicate: (Long) -> Boolean): Long {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun FloatArray.first(predicate: (Float) -> Boolean): Float {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun DoubleArray.first(predicate: (Double) -> Boolean): Double {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun BooleanArray.first(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public inline fun CharArray.first(predicate: (Char) -> Boolean): Char {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
}
```

```java
/**
 * Returns the first element, or `null` if the array is empty.
 */
public fun <T> Array<out T>.firstOrNull(): T? {
    return if (isEmpty())
```
### Open Source Used

The Open Source Used in IOx-FogDirector-AC18 1.17.0 in 2012 includes:

1. **LongArray.indexOf(element: Long): Int**
   
   ```kotlin
   return -1
   ```

2. **IntArray.indexOf(element: Int): Int**
   
   ```kotlin
   for (index in indices) {
   if (element == this[index]) {
   return index
   }
   return -1
   }
   ```

3. **ShortArray.indexOf(element: Short): Int**
   
   ```kotlin
   for (index in indices) {
   if (element == this[index]) {
   return index
   }
   return -1
   }
   ```

4. **ByteArray.indexOf(element: Byte): Int**
   
   ```kotlin
   for (index in indices) {
   if (element == this[index]) {
   return index
   }
   return -1
   }
   ```

5. **T Array.indexOf(element: T): Int**
   
   ```kotlin
   for (index in indices) {
   if (element == this[index]) {
   return index
   }
   return -1
   }
   ```

6. **CharArray.getOrNull(index: Int): Char?**
   
   ```kotlin
   return if (index >= 0 && index <= lastIndex) get(index) else null
   ```

7. **BooleanArray.getOrNull(index: Int): Boolean?**
   
   ```kotlin
   return if (index >= 0 && index <= lastIndex) get(index) else true
   ```

8. **DoubleArray.getOrNull(index: Int): Double?**
   
   ```kotlin
   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
   ```

9. **ByteArray.getOrNull(index: Int): Byte?**
   
   ```kotlin
   return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
   ```

10. **CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

11. **BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

12. **FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

13. **LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

14. **IntArray.getOrElse(index: Int, defaultValue: (Int) -> Int): Int**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

15. **ShortArray.getOrElse(index: Int, defaultValue: (Int) -> Short): Short**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

16. **ByteArray.getOrElse(index: Int, defaultValue: (Int) -> Byte): Byte**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

17. **CharArray.getOrElse(index: Int, defaultValue: (Int) -> Char): Char**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

18. **BooleanArray.getOrElse(index: Int, defaultValue: (Int) -> Boolean): Boolean**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

19. **FloatArray.getOrElse(index: Int, defaultValue: (Int) -> Float): Float**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```

20. **LongArray.getOrElse(index: Int, defaultValue: (Int) -> Long): Long**
    
    ```kotlin
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
    ```
```kotlin
/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun IntArray.indexOfLast(predicate: (Int) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun IntArray.indexOfLast(predicate: (Int) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val lastIndex = indices.lastIndex
        if (predicate(this[lastIndex])) return lastIndex
        return -1
    }
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun ShortArray.indexOfLast(predicate: (Short) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val lastIndex = indices.lastIndex
        if (predicate(this[lastIndex])) return lastIndex
        return -1
    }
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun ByteArray.indexOfLast(predicate: (Byte) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val lastIndex = indices.lastIndex
        if (predicate(this[lastIndex])) return lastIndex
        return -1
    }
}

/**
 * Returns index of the last
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun Array<out T>.indexOfLast(predicate: (T) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun CharArray.indexOfFirst(predicate: (Char) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val firstIndex = indices.firstIndex
        if (predicate(this[firstIndex])) return firstIndex
        return -1
    }
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun BooleanArray.indexOfFirst(predicate: (Boolean) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val firstIndex = indices.firstIndex
        if (predicate(this[firstIndex])) return firstIndex
        return -1
    }
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun DoubleArray.indexOfFirst(predicate: (Double) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val firstIndex = indices.firstIndex
        if (predicate(this[firstIndex])) return firstIndex
        return -1
    }
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun FloatArray.indexOfFirst(predicate: (Float) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val firstIndex = indices.firstIndex
        if (predicate(this[firstIndex])) return firstIndex
        return -1
    }
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public fun LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the first
 * element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun LongArray.indexOfFirst(predicate: (Long) -> Boolean): Int = when {
    indices.isEmpty() -> -1
    else -> {
        val firstIndex = indices.firstIndex
        if (predicate(this[firstIndex])) return firstIndex
        return -1
    }
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2013
element matching the given [predicate], or -1 if the array does not contain such element.

```kotlin
/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun LongArray.indexOfLast(predicate: (Long) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun FloatArray.indexOfLast(predicate: (Float) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun DoubleArray.indexOfLast(predicate: (Double) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun BooleanArray.indexOfLast(predicate: (Boolean) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the array does not contain such element.
 */
public inline fun CharArray.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun <T> Array<out T>.last(): T {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ByteArray.last(): Byte {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun ShortArray.last(): Short {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun IntArray.last(): Int {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun LongArray.last(): Long {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun FloatArray.last(): Float {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun DoubleArray.last(): Double {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun BooleanArray.last(): Boolean {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
public fun CharArray.last(): Char {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return this[lastIndex]
}
```

The text appears to be a part of a source code file, likely containing definitions for various functions that operate on different types of arrays, such as `indexOfLast` and `last`. These functions seem to be part of an open-source library, given the context provided in the metadata.
```kotlin
/**
 * Returns the last element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Array<out T>.lastIndexOf(element: T): Int {
    if (element == null) {
        for (index in indices.reversed()) {
            if (this[index] == null) {
                return index
            }
        }
    } else {
        for (index in indices.reversed()) {
            if (predicate(element) == true) {
                return index
            }
        }
    }
    throw NoSuchElementException("Array contains no element matching the predicate.")
}

/**
 * Returns the last index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == element) {
            return index
        }
    }
    return -1
}

/**
 * Returns the last index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == element) {
            return index
        }
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == element) {
            return index
        }
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == element) {
            return index
        }
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
public fun CharArray.lastIndexOf(element: Char): Int {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (element == element) {
            return index
        }
    }
    return -1
}
```

empty.

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun <T> Array<out T>.lastOrNull(): T? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun ByteArray.lastOrNull(): Byte? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun ShortArray.lastOrNull(): Short? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun IntArray.lastOrNull(): Int? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun LongArray.lastOrNull(): Long? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun FloatArray.lastOrNull(): Float? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun DoubleArray.lastOrNull(): Double? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun BooleanArray.lastOrNull(): Boolean? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element, or `null` if the array is empty.
 */
public fun CharArray.lastOrNull(): Char? {
    return if (isEmpty()) null else this[size - 1]
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun <T, Predicate> Array<out T>.lastOrNull(predicate: Predicate): T? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun ByteArray.lastOrNull(predicate: (Byte) -> Boolean): Byte? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun ShortArray.lastOrNull(predicate: (Short) -> Boolean): Short? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun IntArray.lastOrNull(predicate: (Int) -> Boolean): Int? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun LongArray.lastOrNull(predicate: (Long) -> Boolean): Long? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun FloatArray.lastOrNull(predicate: (Float) -> Boolean): Float? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun DoubleArray.lastOrNull(predicate: (Double) -> Boolean): Double? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun BooleanArray.lastOrNull(predicate: (Boolean) -> Boolean): Boolean? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element was found.
 */
public inline fun CharArray.lastOrNull(predicate: (Char) -> Boolean): Char? {
    for (index in this.indices.reversed()) {
        val element = this[index]
        if (predicate(element)) return element
    }
    return null
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Array<out T>.random(): T {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ByteArray.random(): Byte {
    return random(Random)
}

/**
 * Returns a random element from this array.
 * 
 * @throws NoSuchElementException if this array is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun ShortArray.random(): Short {
    return random(Random)
}
NoSuchElementException if this array is empty.

`IntArray.random()`: Int {`n` return random(Random)`n`}

`LongArray.random()`: Long {`n` return random(Random)`n`}

`FloatArray.random()`: Float {`n` return random(Random)`n`}

`DoubleArray.random()`: Double {`n` return random(Random)`n`}

`CharArray.random()`: Char {`n` return random(Random)`n`}

`ByteArray.random()`: Byte {`n` return get(random.nextInt(size))}

`ShortArray.random()`: Short {`n` return get(random.nextInt(size))}

`IntArray.random(random: Random)`: Int {`n` return get(random.nextInt(size))}

`LongArray.random(random: Random)`: Long {`n` return get(random.nextInt(size))}

`FloatArray.random(random: Random)`: Float {`n` return get(random.nextInt(size))}

`DoubleArray.random(random: Random)`: Double {`n` return get(random.nextInt(size))}

`CharArray.random(random: Random)`: Char {`n` return get(random.nextInt(size))}

`ByteArray.random(random: Random)`: Byte {`n` return get(random.nextInt(size))}

`ShortArray.random(random: Random)`: Short {`n` return get(random.nextInt(size))}

`IntArray.randomOrNull()`: Int? {`n` return randomOrNull(Random)`n`}

`LongArray.randomOrNull()`: Long? {`n` return randomOrNull(Random)`n`}

`FloatArray.randomOrNull()`: Float? {`n` return randomOrNull(Random)`n`}

`DoubleArray.randomOrNull()`: Double? {`n` return randomOrNull(Random)`n`}

`CharArray.randomOrNull()`: Char? {`n` return randomOrNull(Random)`n`}

`ByteArray.randomOrNull()`: Byte? {`n` return randomOrNull(Random)`n`}

`ShortArray.randomOrNull()`: Short? {`n` return randomOrNull(Random)`n`}

`CharArray.randomOrNull(random: Random)`: Char? {`n` return randomOrNull(Random)`n`}

`ByteArray.randomOrNull(random: Random)`: Byte? {`n` return randomOrNull(Random)`n`}

`ShortArray.randomOrNull(random: Random)`: Short? {`n` return randomOrNull(Random)`n`}
random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
ByteArray.randomOrNull(): Byte? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
ShortArray.randomOrNull(): Short? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
IntArray.randomOrNull(): Int? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
LongArray.randomOrNull(): Long? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
FloatArray.randomOrNull(): Float? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
DoubleArray.randomOrNull(): Double? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
\@kotlin.internal.InlineOnly\public inline fun
CharArray.randomOrNull(): Char? { \n    return randomOrNull(Random()) \n} \n\/**
 * Returns a random element from this array, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): T? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Byte? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Short? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Int? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Long? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Float? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Double? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.

*\@SinceKotlin("1.3") @ExperimentalStdlibApi
public fun <T> Array<out T>.randomOrNull(random: Random): Boolean? { \n    if (isEmpty()) \n        return null \n    return get(random.nextInt(size)) \n} \n\/**
 * Returns a random element from this array using the specified source of randomness, or `null` if this array is empty.
element from this array using the specified source of randomness, or `null` if this array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
fun CharArray.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return get(random.nextInt(size))
}

@SinceKotlin("1.3")
@ExperimentalStdlibApi
fun CharArray.randomOrDefault(random: Random, default: Char): Char {
    if (isEmpty())
        return default
    return get(random.nextInt(size))
}
```

`CharArray.randomOrNull()` returns a single element, or throws an exception if the array is empty or has more than one element. It uses a specified source of randomness for generating the random index. If the specified source is `null`, it throws an exception if the array is empty or has more than one element. The method returns `null` if the array is empty. If the array is not empty, it returns the element at the random index.

`CharArray.randomOrDefault()` returns a single element, or provides a default value if the array is empty or has more than one element. It uses a specified source of randomness for generating the random index. If the specified source is `null`, it throws an exception if the array is empty or has more than one element. The method returns the default value if the array is empty. If the array is not empty, it returns the element at the random index.
contains more than one matching element.\n\nsingle = element \nfound = true \n\nif (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
\n@Suppress("UNCHECKED_CAST") \nreturn single as Short? \n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun IntArray.single(predicate: (Int) -> Boolean): Int {
    var single: Int? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Int\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun LongArray.single(predicate: (Long) -> Boolean): Long {
    var single: Long? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Long\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun FloatArray.single(predicate: (Float) -> Boolean): Float {
    var single: Float? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Float\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun DoubleArray.single(predicate: (Double) -> Boolean): Double {
    var single: Double? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Double\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun BooleanArray.single(predicate: (Boolean) -> Boolean): Boolean {
    var single: Boolean? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Boolean\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

inline fun CharArray.single(predicate: (Char) -> Boolean): Char {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    @Suppress("UNCHECKED_CAST") \nreturn single as Char\n* Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element. 

fun <T> Array<out T>.singleOrNull(): T? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

ByteArray.singleOrNull(): Byte? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

IntArray.singleOrNull(): Int? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

LongArray.singleOrNull(): Long? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

CharArray.singleOrNull(): Char? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

ShortArray.singleOrNull(): Short? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

ByteBuffer.singleOrNull(): Byte? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element. 

ArrayOfObject.singleOrNull(): Object? {
    return if (size == 1) this[0] else null
* Returns single element, or ‘null’ if the array is empty or has more than one element.
require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size -
*/
public fun <T> Array<out T>.drop(n: Int): List<T> {

Returns single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.

Returns the single element matching the given [predicate], or `null` if element was not found or more than one element was found.
n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements.\n* \n* \n* \n
* @throws IllegalArgumentException if \[n\] is negative.\n* \n* \n* \n
samples.Collections.Collections.Transformations.dropOne */\n
*public fun LongArray.dropLast(n: Int): List<Long> {\n\nrequire(n >= 0) { \"Requested element count \$n is less than zero.\" }\nreturn take((size - \n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements.\n* \n* \n* \n
* @throws IllegalArgumentException if \[n\] is negative.\n* \n* \n* \n
samples.Collections.Collections.Transformations.dropOne */\n
*public fun FloatArray.dropLast(n: Int): List<Float> {\n\nrequire(n >= 0) { \"Requested element count \$n is less than zero.\" }\nreturn take((size - \n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements.\n* \n* \n* \n
* @throws IllegalArgumentException if \[n\] is negative.\n* \n* \n* \n
samples.Collections.Collections.Transformations.dropOne */\n
*public fun DoubleArray.dropLast(n: Int): List<Double> {\n\nrequire(n >= 0) { \"Requested element count \$n is less than zero.\" }\nreturn take((size - \n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements.\n* \n* \n* \n
* @throws IllegalArgumentException if \[n\] is negative.\n* \n* \n* \n
samples.Collections.Collections.Transformations.dropOne */\n
*public fun BooleanArray.dropLast(n: Int): List<Boolean> {\n\nrequire(n >= 0) { \"Requested element count \$n is less than zero.\" }\nreturn take((size - \n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements.\n* \n* \n* \n
* @throws IllegalArgumentException if \[n\] is negative.\n* \n* \n* \n
samples.Collections.Collections.Transformations.dropOne */\n
*public fun CharArray.dropLast(n: Int): List<Char> {\n\nrequire(n >= 0) { \"Requested element count \$n is less than zero.\" }\nreturn take((size - \n).coerceAtLeast(0))\n|\n|\n
* Returns a list containing all elements except last \[n\] elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun ByteArray.dropLastWhile(predicate: (Byte) -> Boolean): List<Byte> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun ShortArray.dropLastWhile(predicate: (Short) -> Boolean): List<Short> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun IntArray.dropLastWhile(predicate: (Int) -> Boolean): List<Int> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun LongArray.dropLastWhile(predicate: (Long) -> Boolean): List<Long> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun FloatArray.dropLastWhile(predicate: (Float) -> Boolean): List<Float> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */\n
*public fun DoubleArray.dropLastWhile(predicate: (Double) -> Boolean): List<Double> {\n\n\n|\n|\n
* Returns a list containing all elements except last elements that satisfy the given \[predicate\].\n* \n* \n* \n
* @sample samples.collections.Collections.Collections.Transformations.dropOne */
except last elements that satisfy the given \[\text{predicate}\].

```kotlin
public inline fun BooleanArray.dropLastWhile(predicate: (Boolean) -> Boolean): List<Boolean> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}
```

```kotlin
public inline fun CharArray.dropLastWhile(predicate: (Char) -> Boolean): List<Char> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return take(index + 1)
        }
    }
    return emptyList()
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun <T> Array<out T>.dropWhile(predicate: (T) -> Boolean): List<T> {
    var yielding = false
    val list = ArrayList<T>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun ByteArray.dropWhile(predicate: (Byte) -> Boolean): List<Byte> {
    var yielding = false
    val list = ArrayList<Byte>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun ShortArray.dropWhile(predicate: (Short) -> Boolean): List<Short> {
    var yielding = false
    val list = ArrayList<Short>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun IntArray.dropWhile(predicate: (Int) -> Boolean): List<Int> {
    var yielding = false
    val list = ArrayList<Int>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun LongArray.dropWhile(predicate: (Long) -> Boolean): List<Long> {
    var yielding = false
    val list = ArrayList<Long>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun FloatArray.dropWhile(predicate: (Float) -> Boolean): List<Float> {
    var yielding = false
    val list = ArrayList<Float>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```

```kotlin
/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * \[\text{predicate}\].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
public inline fun DoubleArray.dropWhile(predicate: (Double) -> Boolean): List<Double> {
    var yielding = false
    val list = ArrayList<Double>()
    for (item in this)
        if (yielding)
            list.add(item)
        else if (!predicate(item)) {
            list.add(item)
            yielding = true
        }
    return list
}
```
the given [predicate].

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun 

CharArray.filterIndexedTo(predicate: (index: Int, Char) -> Boolean): List<Char> {
    return filterIndexedTo(ArrayList<Char>(), predicate)\n}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> List<T>.filterIndexedTo(destination: C, predicate: (index: Int, T) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <R> List<R>.filterIndexedTo(destination: C, predicate: (index: Int, R) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> ByteArray.filterIndexedTo(destination: C, predicate: (index: Int, Byte) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> ShortArray.filterIndexedTo(destination: C, predicate: (index: Int, Short) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> IntArray.filterIndexedTo(destination: C, predicate: (index: Int, Int) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> LongArray.filterIndexedTo(destination: C, predicate: (index: Int, Long) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> FloatArray.filterIndexedTo(destination: C, predicate: (index: Int, Float) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> DoubleArray.filterIndexedTo(destination: C, predicate: (index: Int, Double) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> BooleanArray.filterIndexedTo(destination: C, predicate: (index: Int, Boolean) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* @param [predicate] function that takes the index of an element and the element itself

and returns the result of predicate evaluation on the element.

* @public inline fun <T> CharArray.filterIndexedTo(destination: C, predicate: (index: Int, Char) -> Boolean): C {
    forEachIndexed { index, element ->
        if (predicate(index, element)) destination.add(element)
    }
    return destination
}

* Returns a list containing all elements that are instances of specified type parameter R.
/*inline fun <reified R> Array<*>.filterIsInstance(): List:@kotlin.internal.NoInfer R> {\n    return filterIsInstanceTo(ArrayList<R遵循>());}/n\n/**\n * Appends all elements that are instances of specified type parameter R to the given [destination].\n */\npublic inline fun <reified R, C : MutableCollection<in R>> Array<*>.filterIsInstanceTo(destination: C): C {\n    for (element in this) if (element is R) destination.add(element)/n\n    return destination\n}/n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n */\npublic inline fun <reified R> Array<*>.filterNot(predicate: (T) -> Boolean): List<T> {\n    return filterNotTo(ArrayList<T遵循>());}/n\n/**\n * Append all elements that are not `null` to the given [destination].\n */\npublic fun <T : Any> Array<out T?>.filterNotNull(): List<T> {\n    return filterNotNullTo(ArrayList<T>())}/n\n/**\n * Appends all elements not matching the given [predicate] to the given [destination].\n */\npublic inline fun <C : MutableCollection<in Byte>> ByteArray.filterNotTo(destination: C, predicate: (Byte) -> Boolean): C {\n    for (element in this) if (!predicate(element)) destination.add(element)/n\n    return destination
}
matching the given \[predicate\] to the given \[destination\].

```kotlin
public inline fun <C : MutableCollection<in Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {
    return destination
}
```

```kotlin
public inline fun <C : Collection<in Boolean>> BooleanArray.filterNotTo(destination: C, predicate: (Boolean) -> Boolean): C {
    for (element in this)
        if (!predicate(element)) destination.add(element)
    return destination
}
```

```kotlin
/**
 * Appends all elements matching the given \[predicate\] to the given \[destination\].
 */
public inline fun <T : Collection<in Boolean>, C : Collection<in T>> T.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this)
        if (predicate(element)) destination.add(element)
    return destination
}
```

```kotlin
/**
 * Appends all elements matching the given \[predicate\] to the given \[destination\].
 */
public inline fun <T, C : Collection<in T>> Array<out T>.filterTo(destination: C, predicate: (T) -> Boolean): C {
    for (element in this)
        if (predicate(element)) destination.add(element)
    return destination
}
```

```kotlin
/**
 * Appends all elements matching the given \[predicate\] to the given \[destination\].
 */
public inline fun <C : Collection<in Byte>> ByteArray.filterTo(destination: C, predicate: (Byte) -> Boolean): C {
    for (element in this)
        if (predicate(element)) destination.add(element)
    return destination
}
```

```kotlin
/**
 * Appends all elements matching the given \[predicate\] to the given \[destination\].
 */
public inline fun <C : Collection<in Char>> CharArray.filterTo(destination: C, predicate: (Char) -> Boolean): C {
    for (element in this)
        if (predicate(element)) destination.add(element)
    return destination
}
```

```kotlin
/**
 * Appends all elements matching the given \[predicate\] to the given \[destination\].
 */
public inline fun <T, C : Collection<in T>> Array<out T>.slice(indices: IntRange): List<T> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun ShortArray.slice(indices: IntRange): List<Short> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun CharArray.slice(indices: IntRange): List<Char> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun IntArray.slice(indices: IntRange): List<Int> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun LongArray.slice(indices: IntRange): List<Long> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun FloatArray.slice(indices: IntRange): List<Float> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```

```kotlin
/**
 * Returns a list containing elements at indices in the specified \[indices\] range.
 */
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    if (indices.isEmpty()) return listOf()
    return copyOfRange(indices.start, indices.endInclusive + 1).asList()
}
```
if (indices.isEmpty()) return listOff();
return copyOfRange(indices.start, indices.endInclusive + 1).asList();

@<T>
public fun BooleanArray.slice(indices: IntRange): List<Boolean> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun ByteArray.slice(indices: IntRange): List<Byte> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun IntArray.slice(indices: IntRange): List<Int> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun ShortArray.slice(indices: IntRange): List<Short> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun LongArray.slice(indices: IntRange): List<Long> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun DoubleArray.slice(indices: IntRange): List<Double> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun CharArray.slice(indices: IntRange): List<Char> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun FloatArray.slice(indices: IntRange): List<Float> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun CharArray.slice(indices: Collection<Int>): List<Char> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun IntArray.slice(indices: Collection<Int>): List<Int> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun ShortArray.slice(indices: Collection<Int>): List<Short> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun LongArray.slice(indices: Collection<Int>): List<Long> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun DoubleArray.slice(indices: Collection<Int>): List<Double> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun CharArray.slice(indices: Collection<Int>): List<Char> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}

@<T>
public fun FloatArray.slice(indices: Collection<Int>): List<Float> {
    return copyOfRange(indices.start, indices.endInclusive + 1).asList();
}
public fun ShortArray.take(n: Int): List<Short> {
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Short>(n)
    for (sourceIndex in indices) {
        list.add(this[sourceIndex])
        if (++count == n)
            break
    }
    return list
}

public fun ByteArray.take(n: Int): List<Byte> {
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Byte>(n)
    for (sourceIndex in indices) {
        list.add(this[sourceIndex])
        if (++count == n)
            break
    }
    return list
}

public fun <T> Array<out T>.take(n: Int): List<T> {
    if (n == 1) return listOf(this[0])
    var count = 0
    var result = emptyList<T>()
    for (sourceIndex in indices) {
        result.add(this[sourceIndex])
        if (++count == n)
            break
    }
    return result
}

public fun CharArray.sliceArray(indices: Collection<Int>): CharArray {
    val result = CharArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

public fun BooleanArray.sliceArray(indices: Collection<Int>): BooleanArray {
    val result = BooleanArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

public fun DoubleArray.sliceArray(indices: Collection<Int>): DoubleArray {
    val result = DoubleArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

public fun FloatArray.sliceArray(indices: Collection<Int>): FloatArray {
    val result = FloatArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}

public fun LongArray.sliceArray(indices: Collection<Int>): LongArray {
    val result = LongArray(indices.size)
    var targetIndex = 0
    for (sourceIndex in indices) {
        result[targetIndex++] = this[sourceIndex]
    }
    return result
}
require(n >= 0) {
    if (n == 0) return emptyList() // Returns a list containing first [n] elements.
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun <T> Array<out T>.take(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun <T> Array<out T>.takeLast(n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun BooleanArray.take(n: Int): List<Boolean> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Boolean>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun ByteArray.take(n: Int): List<Byte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList() // Returns a list containing first [n] elements.
    var count = 0
    val list = ArrayList<Byte>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun LongArray.take(n: Int): List<Long> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Long>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun IntArray.take(n: Int): List<Int> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Int>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun ShortArray.take(n: Int): List<Short> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<Short>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun DoubleArray.take(n: Int): List<Double> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Double>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

private fun FloatArray.take(n: Int): List<Float> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<Float>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

fun take<T>(this: Array<T>, n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

fun takeLast<T>(this: Array<T>, n: Int): List<T> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n >= size) return toList()
    if (n == 1) return listOf(this[size - 1])
    var count = 0
    val list = ArrayList<T>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}
for (index in size - n until size)
    list.add(this[index])
return list

Returns a list containing last [n] elements of an array. If [n] is negative, an `IllegalArgumentException` is thrown.

```java
val list = ArrayList<Char>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Boolean>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Double>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Float>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Long>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Int>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Byte>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<Short>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<ByteArray>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```

```java
val list = ArrayList<T>(n)
for (index in size - n until size)
    list.add(this[index])
return list
```
if (!predicate(this[index])) { return drop(index + 1) } return toList()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Float) -> Boolean): List<Float> { for (index in lastIndex downTo 0) { if (!predicate(this[index])) { return drop(index + 1) } return toList()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Double) -> Boolean): List<Double> { for (index in lastIndex downTo 0) { if (!predicate(this[index])) { return drop(index + 1) } return toList()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Char) -> Boolean): List<Char> { for (index in lastIndex downTo 0) { if (!predicate(this[index])) { return drop(index + 1) } return toList()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Byte) -> Boolean): List<Byte> { val list = ArrayList<Byte>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Short) -> Boolean): List<Short> { val list = ArrayList<Short>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Int) -> Boolean): List<Int> { val list = ArrayList<Int>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Long) -> Boolean): List<Long> { val list = ArrayList<Long>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Float) -> Boolean): List<Float> { val list = ArrayList<Float>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Double) -> Boolean): List<Double> { val list = ArrayList<Double>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Char) -> Boolean): List<Char> { val list = ArrayList<Char>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Byte) -> Boolean): List<Byte> { val list = ArrayList<Byte>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Short) -> Boolean): List<Short> { val list = ArrayList<Short>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Int) -> Boolean): List<Int> { val list = ArrayList<Int>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Long) -> Boolean): List<Long> { val list = ArrayList<Long>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Float) -> Boolean): List<Float> { val list = ArrayList<Float>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Double) -> Boolean): List<Double> { val list = ArrayList<Double>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Char) -> Boolean): List<Char> { val list = ArrayList<Char>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Byte) -> Boolean): List<Byte> { val list = ArrayList<Byte>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Short) -> Boolean): List<Short> { val list = ArrayList<Short>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Int) -> Boolean): List<Int> { val list = ArrayList<Int>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Long) -> Boolean): List<Long> { val list = ArrayList<Long>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Float) -> Boolean): List<Float> { val list = ArrayList<Float>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Double) -> Boolean): List<Double> { val list = ArrayList<Double>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Char) -> Boolean): List<Char> { val list = ArrayList<Char>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Byte) -> Boolean): List<Byte> { val list = ArrayList<Byte>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.

Transformations.takeWhile(predicate: (Short) -> Boolean): List<Short> { val list = ArrayList<Short>() for (item in this) { if (!predicate(item)) break\n
list.add(item)\n
} return list()}\n
* Returns a list containing last elements satisfying the given [predicate].\n
* @sample samples.collections.Collections.
val midPoint = (size / 2) - 1

if (midPoint < 0) return

var reverseIndex = lastIndex

for (index in 0..midPoint) {
    val tmp = this[index]
    this[index] = this[reverseIndex]
    reverseIndex--
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun CharArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun ByteArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun DoubleArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun FloatArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun LongArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun IntArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun ShortArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns an array with elements of this array in reversed order.
 */
public fun BooleanArray.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        reverseIndex--
    }
}

/**
 * Returns a list with elements in reversed order.
 */
public fun List<T>.reversed(): List<T> {
    if (isEmpty()) return emptyList()
    val list = toMutableList()
    list.reverse()
    return list
}

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Array<T>.reversedArray(): Array<T> {
    if (isEmpty()) return this
    val result = arrayOfNulls(this, size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}

/**
 * Returns a list with elements in reversed order.
 */
public fun <T> Array<out T>.reversed(): List<T> {
    if (isEmpty()) return this
    val result = arrayOfNulls(this, size)
    val lastIndex = lastIndex
    for (i in 0..lastIndex)
        result[lastIndex - i] = this[i]
    return result
}
Type: Documentation
CharArray.sortedArrayDescending(): CharArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted according to
 * their natural sort order. 
 */
public fun DoubleArray.sortedArrayDescending(): DoubleArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted according to
 * their natural sort order. 
 */
public fun FloatArray.sortedArrayDescending(): FloatArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted according to
 * their natural sort order. 
 */
public fun LongArray.sortedArrayDescending(): LongArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted according to
 * their natural sort order. 
 */
public fun IntArray.sortedArrayDescending(): IntArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns an array with all elements of this array sorted according to
 * their natural sort order. 
 */
public fun ShortArray.sortedArrayDescending(): ShortArray {
    if (isEmpty()) return this
    return this.copyOf().apply { sortDescending() }
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun CharArray.sortedArray(): List<Char> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun DoubleArray.sorted(): List<Double> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun FloatArray.sorted(): List<Float> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun LongArray.sorted(): List<Long> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun IntArray.sorted(): List<Int> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun ShortArray.sorted(): List<Short> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun ByteArray.sorted(): List<Byte> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun <T : Comparable<T>> Array<T>.sorted(): List<T> {
    return toTypedArray().apply { sort() }.asList()
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
public fun <T : Comparable<T>> ArrayList<T>.sorted(): List<T> {
    return toTypedArray().apply { sort() }.asList()
}
sortDescending() \n\n/**
 * Returns an array with all elements of this array sorted according the specified
 comparator.\n \n * The sort is _stable_. It means that equal elements preserve their order relative to each other
 after sorting.\n */\n
public fun <T> Array<out T>.sortedArrayWith(comparator: Comparator<T>): Array<out T> {\n    if (isEmpty()) return this.copyOf().apply { sortWith(comparator) }\n
    return \n
}\n
/**
 * Returns an array with all elements of this array sorted according the specified
 comparator.\n \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n
public fun <T> Array<out T>.sortedArrayWith(compareBy(comparator: (in T) -> R?): List<T>) {\n    return \n
}\n
/**
 * Returns an array with all elements of this array sorted according the specified
 comparator.\n \n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n */\n
public fun <T> Array<out T>.sortedArrayWith(compareByDescending(comparator: (in T) -> R?): List<T>) {\n    return \n
}\n
/**
 * Returns a list of all elements sorted according to natural sort order of the value returned by specified [selector] function.\n */

ByteArray.sortedBy(crossinline selector: (Byte) -> R?): List<Byte> {\n    return \n
}\n
ShortArray.sortedBy(crossinline selector: (Short) -> R?): List<Short> {\n    return \n
}\n
IntArray.sortedBy(crossinline selector: (Int) -> R?): List<Int> {\n    return \n
}\n
LongArray.sortedBy(crossinline selector: (Long) -> R?): List<Long> {\n    return \n
}\n
FloatArray.sortedBy(crossinline selector: (Float) -> R?): List<Float> {\n    return \n
}\n
DoubleArray.sortedBy(crossinline selector: (Double) -> R?): List<Double> {\n    return \n
}\n
BooleanArray.sortedBy(crossinline selector: (Boolean) -> R?): List<Boolean> {\n    return \n
}\n
CharArray.sortedBy(crossinline selector: (Char) -> R?): List<Char> {\n    return \n
}\n
Array<out T>.sortByDescending(crossinline selector: (in T) -> R?): List<T> {\n    return \n
}\n
*/
according to natural sort order of the value returned by specified \[selector\] function.

```kotlin
public fun <R : Comparable<R>> FloatArray.sortedByDescending(crossinline selector: (Float) -> R?): List<Float> {
    return sortedWith(compareByDescending(selector))
}

public fun <R : Comparable<R>> DoubleArray.sortedByDescending(crossinline selector: (Double) -> R?): List<Double> {
    return sortedWith(compareByDescending(selector))
}

public fun <R : Comparable<R>> BooleanArray.sortedByDescending(crossinline selector: (Boolean) -> R?): List<Boolean> {
    return sortedWith(compareByDescending(selector))
}

public fun <R : Comparable<R>> CharArray.sortedByDescending(crossinline selector: (Char) -> R?): List<Char> {
    return sortedWith(compareByDescending(selector))
}

public fun <T : Comparable<T>> Array<out T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

public fun ByteArray.sortedDescending(): List<Byte> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun ShortArray.sortedDescending(): List<Short> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun IntArray.sortedDescending(): List<Int> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun LongArray.sortedDescending(): List<Long> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun FloatArray.sortedDescending(): List<Float> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun DoubleArray.sortedDescending(): List<Double> {
    return copyOf().apply { sort() }.reversed().toList()
}

public fun BooleanArray.sortedDescending(): List<Boolean> {
    return copyOf().apply { sort() }.reversed().toList()
}
```

The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.
CharArray.sortedWith(comparator: Comparator<in Char>): List<Char> {\n return toTypedArray().apply {
sortWith(comparator) }.asList()\n}\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
<T> Array<out T>.asList(): List<T>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect
fun ByteArray.asList(): List<Byte>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
ShortArray.asList(): List<Short>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
IntArray.asList(): List<Int>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
LongArray.asList(): List<Long>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
FloatArray.asList(): List<Float>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect fun
DoubleArray.asList(): List<Double>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic expect
fun BooleanArray.asList(): List<Boolean>\n\n/**\n * Returns a [List] that wraps the original array.\n */\npublic
expect fun CharArray.asList(): List<Char>\n\n/**\n * Returns `true` if the two specified arrays are *deeply* equal
to one another,\n * i.e. contain the same number of the same elements in the same order.\n * \n * If two
corresponding elements are nested arrays, they are also compared deeply.\n * If any of arrays contains itself on any
nesting level the behavior is undefined.\n * \n * The elements of other types are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun <T> Array<out T>.contentDeepEquals(other:
Array<out T>): Boolean\n\n/**\n * Returns a hash code based on the contents of this array as if it is [List].\n *
Nested arrays are treated as lists too.\n * \n * If any of arrays contains itself on any nesting level the behavior is
undefined.\n */\n@SinceKotlin(\"1.1\")\npublic expect fun <T> Array<out T>.contentDeepHashCode(): Int\n\n/**\n
* Returns a string representation of the contents of this array as if it is a [List].\n * Nested arrays are treated as lists
too.\n * \n * If any of arrays contains itself on any nesting level that reference\n * is rendered as `\"[...]\"` to prevent
recursion.\n * \n * @sample samples.collections.Arrays.ContentOperations.contentDeepToString\n
*/\n@SinceKotlin(\"1.1\")\npublic expect fun <T> Array<out T>.contentDeepToString(): String\n\n/**\n * Returns
`true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain the same number of the
same elements in the same order.\n * \n * The elements are compared for equality with the [equals][Any.equals]
function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`.\n
*/\n@SinceKotlin(\"1.1\")\npublic expect infix fun <T> Array<out T>.contentEquals(other: Array<out T>):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain
the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun ByteArray.contentEquals(other: ByteArray):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain
the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun ShortArray.contentEquals(other: ShortArray):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain
the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun IntArray.contentEquals(other: IntArray):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain
the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun LongArray.contentEquals(other: LongArray):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain
the same number of the same elements in the same order.\n * \n * The elements are compared for equality with the
[equals][Any.equals] function.\n * For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not
equal to `0.0`.\n */\n@SinceKotlin(\"1.1\")\npublic expect infix fun FloatArray.contentEquals(other: FloatArray):
Boolean\n\n/**\n * Returns `true` if the two specified arrays are *structurally* equal to one another,\n * i.e. contain

Open Source Used In IOx-FogDirector-AC18 1.17.0 2039


the same number of the same elements in the same order. The elements are compared for equality with the `equals` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. The elements are compared for equality with the `[equals]` function. For floating point numbers it means that `NaN` is equal to itself and `-0.0` is not equal to `0.0`. The elements are compared for equality with the `[equals]` function.

```kotlin
@SinceKotlin("1.1")
public expect infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean
```

```kotlin
@SinceKotlin("1.1")
public expect infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean
```

```kotlin
@SinceKotlin("1.1")
public expect infix fun CharArray.contentEquals(other: CharArray): Boolean
```

```kotlin
@SinceKotlin("1.1")
public expect fun <T> Array<out T>.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun ByteArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun ShortArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun IntArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun LongArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun FloatArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun DoubleArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun BooleanArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun CharArray.contentHashCode(): Int
```

```kotlin
@SinceKotlin("1.1")
public expect fun <T> Array<out T>.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun ByteArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun ShortArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun IntArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun LongArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun FloatArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun DoubleArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun BooleanArray.contentToString(): String
```

```kotlin
@SinceKotlin("1.1")
public expect fun CharArray.contentToString(): String
```
Open Source Used In IOx-FogDirector-AC18 1.17.0 2041

its subrange into the [destination] array and returns that array. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. @param destination the array to copy to. @param destinationOffset the position in the [destination] array to copy to, 0 by default. @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default. @param endIndex the end (exclusive) of the subrange to copy, size of this array by default. *

IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range. @return the [destination] array.

Expect fun <T> Array<T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> Copies this array or its subrange into the [destination] array and returns that array. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range. @return the [destination] array.

public expect fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

Expect fun <T> ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray Copies this array or its subrange into the [destination] array and returns that array. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range. @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`. @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range. @return the [destination] array.

public expect fun IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray

Expect fun <T> ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray

Expect fun <T> ShortArray.copyInto(destination: ShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ShortArray

Expect fun <T> IntArray.copyInto(destination: IntArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): IntArray
Open Source Used In IOx-FogDirector-AC18 1.17.0 2042
```kotlin
public expect fun <T> Array<T>.copyOf():
    Array<T>

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun ByteArray.copyOf(): ByteArray

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun ShortArray.copyOf(): ShortArray

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun IntArray.copyOf(): IntArray

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun LongArray.copyOf(): LongArray

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun FloatArray.copyOf(): FloatArray

/**
 * Returns new array which is a copy of the original array.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public expect fun DoubleArray.copyOf(): DoubleArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun ByteArray.copyOf(newSize: Int): ByteArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun ShortArray.copyOf(newSize: Int): ShortArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun IntArray.copyOf(newSize: Int): IntArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun LongArray.copyOf(newSize: Int): LongArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun FloatArray.copyOf(newSize: Int): FloatArray

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * 
 * - If [newSize] is less than the size of the original array, the copy array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * 
 * @sample samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf
 */
public expect fun DoubleArray.copyOf(newSize: Int): DoubleArray
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2043
copy of the original array, resized to the given [newSize].
The copy is either truncated or padded at the end with
`false` values if necessary.
- If [newSize] is less than the size of the original array, the copy array is
  truncated to the [newSize].
- If [newSize] is greater than the size of the original array, the extra elements in the
  copy array are filled with `false` values.

```kotlin
samples.collections.Arrays.CopyOfOperations.resizedPrimitiveCopyOf

public expect fun BooleanArray.copyOf(newSize: Int): BooleanArray
```

```kotlin
public expect fun CharArray.copyOf(newSize: Int): CharArray
```

```kotlin
public expect fun <T> Array<T>.copyOf(newSize: Int): Array<T?>
```

```kotlin
public expect fun <T> Array<T>.copyOfRange(fromIndex: Int, toIndex: Int): Array<T>
```

```kotlin
public expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray
```

```kotlin
public expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray
```

```kotlin
public expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray
```

```kotlin
public expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray
```

```kotlin
public expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray
```

```kotlin
public expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray
```

```kotlin
public expect fun BooleanArray.copyOfRange(fromIndex: Int, toIndex: Int): BooleanArray
```

```kotlin
CharArray.copyOfRange(fromIndex: Int): CharArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun ByteArray.copyOfRange(fromIndex: Int, toIndex: Int): ByteArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun ShortArray.copyOfRange(fromIndex: Int, toIndex: Int): ShortArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun FloatArray.copyOfRange(fromIndex: Int, toIndex: Int): FloatArray
```

```kotlin
@Suppress("NO_ACTUAL_FOR_EXPECT")
public expect fun DoubleArray.copyOfRange(fromIndex: Int, toIndex: Int): DoubleArray
```

```kotlin
charArray.copyOfRange(fromIndex: Int): CharArray
```
Returns the range of valid indices for the array.

```kotlin
*\n\n\n@\n\n\ninternal.InlineOnly\n\n\n\ninline fun <T> Array<out T>.isNotEmpty(): Boolean {
    return size != 0
}
```

```kotlin
*\n\n\nArray<out T>.isEmpty(): Boolean {
    return size == 0
}
```

```kotlin
*\n\n\nArray<out T>.isNotEmpty(): Boolean {
    return !isEmpty()
}
```

```kotlin
*\n\n\nArray<out T>.lastIndex: Int
    get() = size - 1
```

```kotlin
*\n\n\nByteArray.isEmpty(): Boolean {
    return size == 0
}
```

```kotlin
*\n\n\nByteArray.isNotEmpty(): Boolean {
    return !isEmpty()
}
```

```kotlin
*\n\n\nByteArray.lastIndex: Int
    get() = size - 1
```

```kotlin
*\n\n\nCharArray.isEmpty(): Boolean {
    return size == 0
}
```

```kotlin
*\n\n\nCharArray.isNotEmpty(): Boolean {
    return !isEmpty()
}
```

```kotlin
*\n\n\nCharArray.lastIndex: Int
    get() = size - 1
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```

```kotlin
*\n\n\noperator fun <T> Array<T>.plus(element: T): Array<T>
```

```kotlin
*\n\n\noperator fun ByteArray.plus(element: Byte): ByteArray
```
elements of the original array and then the given [element].\n */\npublic expect operator fun ShortArray.plus(element: Short): ShortArray\n
/**
 * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun IntArray.plus(element: Int): IntArray

/**
 * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun LongArray.plus(element: Long): LongArray

/**
 * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun FloatArray.plus(element: Float): FloatArray

/**
 * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun DoubleArray.plus(element: Double): DoubleArray

/**
 * Returns an array containing all elements of the original array and then the given [element].\n */\npublic expect operator fun BooleanArray.plus(element: Boolean): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> ByteArray.plus(elements: Collection<Byte>): ByteArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> ShortArray.plus(elements: Collection<Short>): ShortArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> IntArray.plus(elements: Collection<Int>): IntArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> LongArray.plus(elements: Collection<Long>): LongArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> FloatArray.plus(elements: Collection<Float>): FloatArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> DoubleArray.plus(elements: Collection<Double>): DoubleArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> BooleanArray.plus(elements: Collection<Boolean>): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> CharArray.plus(elements: Collection<Char>): CharArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> Array<T>.plus(elements: Collection<T>): Array<T>

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> ByteArray.plus(elements: Collection<Byte>): ByteArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> ShortArray.plus(elements: Collection<Short>): ShortArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> IntArray.plus(elements: Collection<Int>): IntArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> LongArray.plus(elements: Collection<Long>): LongArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> FloatArray.plus(elements: Collection<Float>): FloatArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> DoubleArray.plus(elements: Collection<Double>): DoubleArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> BooleanArray.plus(elements: Collection<Boolean>): BooleanArray

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.\n */\npublic expect operator fun <T> CharArray.plus(elements: Collection<Char>): CharArray
array and then the given [element]. */
plusElement(element: T): Array<T> = @sample

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortIntArray
 */
public expect fun IntArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortLongArray
 */
public expect fun LongArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortByteArray
 */
public expect fun ByteArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortCharArray
 */
public expect fun CharArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortDoubleArray
 */
public expect fun DoubleArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortFloatArray
 */
public expect fun FloatArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortShortArray
 */
public expect fun ShortArray.sort(): Unit

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArrayOfComparable
 */
public expect fun <T : Comparable<T>> Array<out T>.sort(): Unit

/**
 * Returns an array of Boolean containing all of the elements of this generic array.
 */
public expect fun ByteArray.toBooleanArray(): Array<Byte>

/**
 * Returns an array of Byte containing all of the elements of this generic array.
 */
public expect fun ShortArray.toByteArray(): Array<Short>

/**
 * Returns an array of Int containing all of the elements of this generic array.
 */
public expect fun IntArray.toIntArray(): Array<Int>

/**
 * Returns an array of Long containing all of the elements of this generic array.
 */
public expect fun LongArray.toLongArray(): Array<Long>

/**
 * Returns an array of Float containing all of the elements of this generic array.
 */
public expect fun FloatArray.toFloatArray(): Array<Float>

/**
 * Returns an array of Double containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toDoubleArray(): Array<Double>

/**
 * Returns an array of Char containing all of the elements of this primitive array.
 */
public expect fun CharArray.toCharArray(): Array<Char>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ByteArray.toTypedArray(): Array<Byte>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun ShortArray.toTypedArray(): Array<Short>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun IntArray.toTypedArray(): Array<Int>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun LongArray.toTypedArray(): Array<Long>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun FloatArray.toTypedArray(): Array<Float>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun DoubleArray.toTypedArray(): Array<Double>

/**
 * Returns a *typed* object array containing all of the elements of this primitive array.
 */
public expect fun CharArray.toTypedArray(): Array<Char>

/**
 * Returns a [Map] containing key-value pairs provided by [transform] function applied to elements of the given array.
 */

Open Source Used In IOx-FogDirector-AC18 1.17.0 2048
Array<out T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

ByteArray.associate(transform: (Byte) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

ShortArray.associate(transform: (Short) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

IntArray.associate(transform: (Int) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

LongArray.associate(transform: (Long) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

FloatArray.associate(transform: (Float) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

DoubleArray.associate(transform: (Double) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}

BooleanArray.associate(transform: (Boolean) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)/*
    * Returns a [Map] containing key-value pairs provided by [transform] function
    * applied to elements of the given array.
    * If any of two pairs would have the same key the last one gets added to the
    * returned map.
    * The returned map preserves the entry iteration order of the original array.
    * @sample samples.collections.Arrays.Transformations.associateArrayOfPrimitives
    */
}
CharArray.associate(transform: (Char) -> Pair<K, V>): Map<K, V>  
    val capacity = 
    mapCapacity(size).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), 
    transform)
  **\n  * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <T, K> 
  * Array<out T>.associateBy(keySelector: (T) -> K): Map<K, T>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, T>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <K> 
  * ByteArray.associateBy(keySelector: (Byte) -> K): Map<K, Byte>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, Byte>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <K> 
  * ShortArray.associateBy(keySelector: (Short) -> K): Map<K, Short>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, Short>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <K> 
  * IntArray.associateBy(keySelector: (Int) -> K): Map<K, Int>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, Int>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <K> 
  * LongArray.associateBy(keySelector: (Long) -> K): Map<K, Long>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, Long>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample 

samples.collections.Arrays Transformations associateArrayOfPrimitivesBy
  * /public inline fun <K> 
  * FloatArray.associateBy(keySelector: (Float) -> K): Map<K, Float>  
  * val capacity = 
  * mapCapacity(size).coerceAtLeast(16)
  * return associateByTo(LinkedHashMap<K, Float>(capacity), 
  * keySelector)
  * * Returns a [Map] containing the elements from the given array indexed by the key 
  * returned from [keySelector] function applied to each element. 
  * If any two elements would have the same key returned by [keySelector] the last one gets added to the map. 
  * The returned map preserves the entry iteration order of the original array. 
  * @sample
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesBy

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>
  DoubleArray.associateBy(keySelector: (Double) -> K): Map<K, Double> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Double>(capacity),
      keySelector, { key: K -> key })
}

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>
  BooleanArray.associateBy(keySelector: (Boolean) -> K): Map<K, Boolean> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Boolean>(capacity),
      keySelector, { key: K -> key })
}

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K>
  CharArray.associateBy(keySelector: (Char) -> K): Map<K, Char> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, Char>(capacity),
      keySelector, { key: K -> key })
}

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V>
  ByteArray.associateBy(keySelector: (Byte) -> K, valueTransform: (Byte) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
      keySelector, valueTransform)
}

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V>
  ShortArray.associateBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
      keySelector, valueTransform)
}

/**
 * Returns a [Map] containing the elements from the given array indexed by the key.
 * If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original array.
 */
public inline fun <K, V>
  IntArray.associateBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
      keySelector, valueTransform)
}
the entry iteration order of the original array.

```kotlin
fun <K, V> LongArray.associateBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector, valueTransform)  // Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
}

fun <K, V> FloatArray.associateBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector, valueTransform)  // Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
}

fun <K, V> DoubleArray.associateBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector, valueTransform)  // Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
}

fun <K, V> BooleanArray.associateBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector, valueTransform)  // Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
}

fun <K, V> CharArray.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(size).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity),
            keySelector, valueTransform)  // Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given array.
}

fun <T, K, M : MutableMap<K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)  // Populates and returns the [destination] mutable map with key-value pairs, and value is the element itself.
    }
    return destination
}
```

If any two elements would have the same key returned by [keySelector], the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

```kotlin
fun <T, K, M : MutableMap<K, in T>> ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)  // Populates and returns the [destination] mutable map with key-value pairs, and value is the element itself.
    }
    return destination
}
```

If any two elements would have the same key returned by [keySelector], the last one gets added to the map. The returned map preserves the entry iteration order of the original array.

```kotlin
fun <T, K, M : MutableMap<K, in T>> Array<out T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)  // Populates and returns the [destination] mutable map with key-value pairs, and value is the element itself.
    }
    return destination
}
```

If any two elements would have the same key returned by [keySelector], the last one gets added to the map. The returned map preserves the entry iteration order of the original array.
public inline fun <K, M : MutableMap<in K, in Short>> ShortArray.associateByTo(destination: M, keySelector: (Short) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function applied to each element of the given array and value is the element itself. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <K, M : MutableMap<in K, in Int>> IntArray.associateByTo(destination: M, keySelector: (Int) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function applied to each element of the given array and value is the element itself. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <K, M : MutableMap<in K, in Long>> LongArray.associateByTo(destination: M, keySelector: (Long) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function applied to each element of the given array and value is the element itself. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <K, M : MutableMap<in K, in Float>> FloatArray.associateByTo(destination: M, keySelector: (Float) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function applied to each element of the given array and value is the element itself. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <K, M : MutableMap<in K, in Double>> DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map.
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateByToWithValueTransform(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}
the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. * @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > ByteArray.associateByTo(destination: M, keySelector: (Byte) -> K, valueTransform: (Byte) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > ShortArray.associateByTo(destination: M, keySelector: (Short) -> K, valueTransform: (Short) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > IntArray.associateByTo(destination: M, keySelector: (Int) -> K, valueTransform: (Int) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > LongArray.associateByTo(destination: M, keySelector: (Long) -> K, valueTransform: (Long) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > FloatArray.associateByTo(destination: M, keySelector: (Float) -> K, valueTransform: (Float) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > DoubleArray.associateByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

| K, V, M : MutableMap<in K, in V> > BooleanArray.associateByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M |
| for (element in this) |
| destination.put(keySelector(element), valueTransform(element)) |
| return destination[

/* Populates and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. */ @sample 

samples.collections.Arrays.Transformations.associateArrayOfPrimitivesByToWithValueTransform

and returns the [destination] mutable map with key-value pairs, where key is provided by the [keySelector] function and value is provided by the [valueTransform] function applied to elements of the given array. If any two elements would have the same key returned by [keySelector] the last one gets added to the map. Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any two pairs would have the same key the last one gets added to the map. Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array. If any two pairs would have the same key the last one gets added to the map.

```kotlin
public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Array<out T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <K, V, M : MutableMap<in K, in V>> ByteArray.associateTo(destination: M, transform: (Byte) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <K, V, M : MutableMap<in K, in V>> ShortArray.associateTo(destination: M, transform: (Short) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <K, V, M : MutableMap<in K, in V>> IntArray.associateTo(destination: M, transform: (Int) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <K, V, M : MutableMap<in K, in V>> FloatArray.associateTo(destination: M, transform: (Float) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 */
public inline fun <K, V, M : MutableMap<in K, in V>> DoubleArray.associateTo(destination: M, transform: (Double) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}
```
samples.collections.Arrays.Transformations.associateArrayOfPrimitivesTo

*/

public inline fun <K, V, M : MutableMap<in K, in V>> BooleanArray.associateTo(destination: M, transform: (Boolean) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given array.
 * If any of two pairs would have the same key the last one gets added to the map.
 */

public inline fun <K, V, M : MutableMap<in K, in V>> CharArray.associateTo(destination: M, transform: (Char) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <T, C : MutableCollection<in T>> Array<out T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Byte>> ByteArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Short>> ShortArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Int>> IntArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Long>> LongArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Float>> FloatArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Double>> DoubleArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Boolean>> BooleanArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 */

public fun <C : MutableCollection<in Char>> CharArray.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun <T> Array<out T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun ByteArray.toHashSet(): HashSet<Byte> {
    return toCollection(HashSet<Byte>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun ShortArray.toHashSet(): HashSet<Short> {
    return toCollection(HashSet<Short>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun IntArray.toHashSet(): HashSet<Int> {
    return toCollection(HashSet<Int>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun LongArray.toHashSet(): HashSet<Long> {
    return toCollection(HashSet<Long>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun FloatArray.toHashSet(): HashSet<Float> {
    return toCollection(HashSet<Float>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun DoubleArray.toHashSet(): HashSet<Double> {
    return toCollection(HashSet<Double>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun BooleanArray.toHashSet(): HashSet<Boolean> {
    return toCollection(HashSet<Boolean>(mapCapacity(size)))
}

/**
 * Returns a [HashSet] of all elements.
 */

public fun CharArray.toHashSet(): HashSet<Char> {
    return toCollection(HashSet<Char>(mapCapacity(size)))
}

/**
 * Returns a [List] containing all elements.
 */

public fun <T> Array<out T>.toList(): List<T> {
    return when (size) {
        0 -> emptyList()
        1 -> listOf(this[0])
        else -> this.toMutableList()
    }
}
*\n\n\n```kotlin
*\n\n\n```
toCollection(LinkedHashSet<Long>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun DoubleArray.toSet(): Set<Double> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Double>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun FloatArray.toSet(): Set<Float> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Float>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun IntArray.toSet(): Set<Int> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Int>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun LongArray.toSet(): Set<Long> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Long>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun ShortArray.toSet(): Set<Short> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Short>(mapCapacity(size)))
    }
}

/**
 * Returns a [Set] of all elements.
 */
public fun BooleanArray.toSet(): Set<Boolean> {
    return when (size) {
        0 -> emptySet()
        1 -> setOf(this[0])
        else ->
            toCollection(LinkedHashSet<Boolean>(mapCapacity(size)))
    }
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> Collection<Array<out T>> flatMap(transform: (T) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> ByteArray.flatMap(transform: (Byte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> ShortArray.flatMap(transform: (Short) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> IntArray.flatMap(transform: (Int) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> LongArray.flatMap(transform: (Long) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> FloatArray.flatMap(transform: (Float) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> DoubleArray.flatMap(transform: (Double) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> BooleanArray.flatMap(transform: (Boolean) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.
 */
public inline fun <R> CharArray.flatMap(transform: (Char) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

// For element in this
val list = transform(element)
destination.addAll(list);\n
return destination;\n\n/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> ByteArray.flatMapTo(destination: C, transform: (Byte) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> IntArray.flatMapTo(destination: C, transform: (Int) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> LongArray.flatMapTo(destination: C, transform: (Long) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].*/

public inline fun <R, C : MutableCollection<R>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <T, K> Array<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector);

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> ShortArray.flatMapTo(destination: C, transform: (Short) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> FloatArray.flatMapTo(destination: C, transform: (Float) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> DoubleArray.flatMapTo(destination: C, transform: (Double) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> BooleanArray.flatMapTo(destination: C, transform: (Boolean) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;

/* Groups elements of the original array by the key returned by the given [keySelector] function.*/

public inline fun <K, MutableList<T>> CharArray.flatMapTo(destination: C, transform: (Char) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element);\n        destination.addAll(list);\n    }
    return destination;
where each group key is associated with a list of corresponding elements.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.

The returned map preserves the entry iteration order of the keys produced from the original array.
with a list of corresponding values. The returned map preserves the entry iteration order of the keys produced from the original array.

```kotlin
samples.collections.Collections.Transformations.groupByKeysAndValues

public inline fun <K, V> ShortArray.groupBy(keySelector: (Short) -> K, valueTransform: (Short) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 * @sample
 * samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V> IntArray.groupBy(keySelector: (Int) -> K, valueTransform: (Int) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 * @sample
 * samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V> LongArray.groupBy(keySelector: (Long) -> K, valueTransform: (Long) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 * @sample
 * samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V> FloatArray.groupBy(keySelector: (Float) -> K, valueTransform: (Float) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 * @sample
 * samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V> DoubleArray.groupBy(keySelector: (Double) -> K, valueTransform: (Double) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element * and returns a map where each group key is associated with a list of corresponding values.
 * The returned map preserves the entry iteration order of the keys produced from the original array.
 * @sample
 * samples.collections.Collections.Transformations.groupByKeysAndValues
 */
public inline fun <K, V> BooleanArray.groupBy(keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to
the [destination] map each group key associated with a list of corresponding elements.\n* \n* @return The [destination] map.
* 
* @sample samples.collections.Collections.Transformations.groupBy\n* \n* @public inline
* fun <T, K : MutableMap<in K, MutableList<T>>, M : MutableList<MutableList<T>>> Array<out T>.groupByTo(destination: M, keySelector: (T) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<T>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<ByteArray>>> ByteArray.groupByTo(destination: M, keySelector: (ByteArray) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<ByteArray>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Short>>>'"Short'.toInt()".groupByTo(destination: M, keySelector: (Short) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Short>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Int>>> IntArray.groupByTo(destination: M, keySelector: (Int) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Int>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Long>>>'"Long'.toLong()".groupByTo(destination: M, keySelector: (Long) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Long>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Float>>> FloatArray.groupByTo(destination: M, keySelector: (Float) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Float>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Double>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Double>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
* 
* @public inline
* fun <K, M : MutableMap<in K, MutableList<Boolean>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K): M {
*     for (element in this) {
*         val key = keySelector(element)
*         val list = destination.getOrPut(key) { ArrayList<Boolean>() }
*         list.add(element)
*         return destination
*     }
*     return destination
* }
*
Open Source Used In IOX-FogDirector-AC18 1.17.0 2063

function\n * applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupBy\n */\n
public inline fun <K, M : MutableMap<in K,\nvalueSelector: (Char) -> K):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<Char>()\n        }\n        list.add(element)\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <T, V, M :\nvalueSelector: (T) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <K, V, M :\nvalueSelector: (Byte) -> K,\nvalueTransform: (Byte) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <K, V, M :\nvalueSelector: (Short) -> K,\nvalueTransform: (Short) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <K, V, M :\nvalueSelector: (Int) -> K,\nvalueTransform: (Int) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <K, V, M :\nvalueSelector: (Long) -> K,\nvalueTransform: (Long) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */\n
public inline fun <K, V, M :\nvalueSelector: (Float) -> K,\nvalueTransform: (Float) -> V):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<V>()\n        }\n        list.add(valueTransform(element))\n    }\n    return destination\n
/**\n * Groups values returned by the [valueTransform] function applied to each element of the original array\n * by the key returned by the given [keySelector] function applied to the element\n * and puts to the [destination] map each group key associated with a list of corresponding values.\n * @return The [destination] map.\n * @sample

samples.collections.Collections.Transformations.groupByKeysAndValues\n */

public inline fun <K, M :\nvalueSelector: (Char) -> K):\n    for (element in this) {\n        val key = keySelector(element)\n        val list = destination.getOrPut(key) {\n            ArrayList<Char>()\n        }\n        list.add(element)\n    }\n    return destination
*/

destination

* Groups values returned by the [valueTransform] function applied to each element of the original array

* by the key returned by the given [keySelector] function applied to the element

* and puts to the [destination] map each group key associated with a list of corresponding values.

* @return The [destination] map

* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

* @public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> DoubleArray.groupByTo(destination: M, keySelector: (Double) -> K, valueTransform: (Double) -> V): M {

    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

* Groups values returned by the [valueTransform] function applied to each element of the original array

* by the key returned by the given [keySelector] function applied to the element

* and puts to the [destination] map each group key associated with a list of corresponding values.

* @return The [destination] map

* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

* @public inline fun <K, V, M : MutableMap<in K, MutableList<V>>> BooleanArray.groupByTo(destination: M, keySelector: (Boolean) -> K, valueTransform: (Boolean) -> V): M {

    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

* Creates a [Grouping] source from an array to be used later with one of group-and-fold operations

* using the specified [keySelector] function to extract a key from each element.

* @sample samples.collections.Collections.Transformations.groupByKeysAndValues

* @public inline fun <T, K> Array<out T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {

    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <T, R> Array<out T>.map(transform: (T) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> ByteArray.map(transform: (Byte) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> ShortArray.map(transform: (Short) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> IntArray.map(transform: (Int) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> LongArray.map(transform: (Long) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> FloatArray.map(transform: (Float) -> R): List<R> {

    return mapTo(ArrayList<R>(size), transform)
}

* Returns a list containing the results of applying the given [transform] function

* to each element in the original array.

* @sample samples.collections.Collections.Transformations.map

* @public inline fun <R> DoubleArray.map(transform:...
(Double) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform) // Returns a list containing the results of applying the given [transform] function to each element in the original array.
}

/**
 * Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.mapIndexedTo
 */
public inline fun <T, R> Array<out T>.mapIndexedTo(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform) // Returns a list containing the results of applying the given [transform] function to each element and its index in the original array.
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.mapIndexedTo
 */
public inline fun <T, R : Any> Array<out T>.mapIndexedToNonNull(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedToNonNull(ArrayList<R>(size), transform) // Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.mapIndexedToNonNull
 */
public inline fun <T, R> CharArray.mapIndexedToNonNull(transform: (index: Int, Char) -> R): List<R> {
    return mapIndexedToNonNull(ArrayList<R>(size), transform) // Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.
 * @sample samples.collections.Collections.Transformations.mapIndexedToNonNull
 */
public inline fun <T, R> CharArray.mapIndexedToNonNull(transform: (index: Int, Char) -> R): List<R> {
    return mapIndexedToNonNull(ArrayList<R>(size), transform) // Returns a list containing only the non-null results of applying the given [transform] function to each element and its index in the original array.
}
T.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R>

R>> Array<out T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C

ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C

IntArray.mapIndexedTo(destination: C, transform: (index: Int, Int) -> R): C

LongArray.mapIndexedTo(destination: C, transform: (index: Int, Long) -> R): C

ShortArray.mapIndexedTo(destination: C, transform: (index: Int, Short) -> R): C

ByteArray.mapIndexedTo(destination: C, transform: (index: Int, Byte) -> R): C

DoubleArray.mapIndexedTo(destination: C, transform: (index: Int, Double) -> R): C

FloatArray.mapIndexedTo(destination: C, transform: (index: Int, Float) -> R): C

BooleanArray.mapIndexedTo(destination: C, transform: (index: Int, Boolean) -> R): C
[destination].

* @param [transform] function that takes the index of an element and the element itself\n and returns the result of the transform applied to the element.\n */
public inline fun <R, C : MutableCollection<in R>> CharArray.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element in the original array.\n */
public inline fun <T, R : Any> Array<out T>.mapNotNull(transform: (T) -> R?): List<R> {
    var forEach = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function\n * to each element in the original array.\n */
public inline fun <T, R : Any> Array<out T>.mapNotNullTo(destination: List<R>, transform: (T) -> R?): List<R> {
    var forEach = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :.MutableCollection<in R>> Array<out R>.mapTo(destination: C, transform: (R) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> ByteArray.mapTo(destination: C, transform: (Byte) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> ShortArray.mapTo(destination: C, transform: (Short) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> IntArray.mapTo(destination: C, transform: (Int) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> LongArray.mapTo(destination: C, transform: (Long) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> FloatArray.mapTo(destination: C, transform: (Float) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> DoubleArray.mapTo(destination: C, transform: (Double) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> BooleanArray.mapTo(destination: C, transform: (Boolean) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array\n and appends the results to the given [destination].\n */
public inline fun <R, C :MutableCollection<in R>> CharArray.mapTo(destination: C, transform: (Char) -> R): C {
    var for Each = false
    for (item in this)
        if (item != null)
            destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun <T> Array<out T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun ByteArray.withIndex(): Iterable<IndexedValue<Byte>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun ShortArray.withIndex(): Iterable<IndexedValue<Short>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array\n * into an [IndexedValue] containing the index of that element and the element itself.\n */
public fun IntArray.withIndex(): Iterable<IndexedValue<Int>>
public fun LongArray.withIndex(): Iterable<IndexedValue<Long>> {
    return IndexingIterable { iterator() }
}

public fun FloatArray.withIndex(): Iterable<IndexedValue<Float>> {
    return IndexingIterable { iterator() }
}

public fun DoubleArray.withIndex(): Iterable<IndexedValue<Double>> {
    return IndexingIterable { iterator() }
}

public fun BooleanArray.withIndex(): Iterable<IndexedValue<Boolean>> {
    return IndexingIterable { iterator() }
}

public fun CharArray.withIndex(): Iterable<IndexedValue<Char>> {
    return IndexingIterable { iterator() }
}

public fun <T> Array<out T>.distinct(): List<T> {
    return this.toMutableSet().toList()
}

public fun ByteArray.distinct(): List<Byte> {
    return this.toMutableSet().toList()
}

public fun ShortArray.distinct(): List<Short> {
    return this.toMutableSet().toList()
}

public fun IntArray.distinct(): List<Int> {
    return this.toMutableSet().toList()
}

public fun LongArray.distinct(): List<Long> {
    return this.toMutableSet().toList()
}

public fun FloatArray.distinct(): List<Float> {
    return this.toMutableSet().toList()
}

public fun DoubleArray.distinct(): List<Double> {
    return this.toMutableSet().toList()
}

public fun BooleanArray.distinct(): List<Boolean> {
    return this.toMutableSet().toList()
}

public fun CharArray.distinct(): List<Char> {
    return this.toMutableSet().toList()
}
T>.distinctBy(selector: (T) -> K): List<T> {
    val set = HashSet<K>()
    val list = ArrayList<T>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array with distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy

ByteArray.distinctBy(selector: (Byte) -> K): List<Byte> {
    val set = HashSet<K>()
    val list = ArrayList<Byte>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array with distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy

ShortArray.distinctBy(selector: (Short) -> K): List<Short> {
    val set = HashSet<K>()
    val list = ArrayList<Short>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array with distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy

IntArray.distinctBy(selector: (Int) -> K): List<Int> {
    val set = HashSet<K>()
    val list = ArrayList<Int>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array with distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy

LongArray.distinctBy(selector: (Long) -> K): List<Long> {
    val set = HashSet<K>()
    val list = ArrayList<Long>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}

/**
 * Returns a list containing only elements from the given array with distinct keys returned by the given [selector] function.
 * The elements in the resulting list are in the same order as they were in the source array.
 */
@sample samples.collections.Collections.Transformations.distinctAndDistinctBy

CharArray.distinctBy(selector: (Char) -> K): List<Char> {
    val set = HashSet<K>()
    val list = ArrayList<Char>()
    for (e in this) {
        val key = selector(e)
        if (set.add(key))
            list.add(e)
    }
    return list
}
Open Source Used In IOx-FogDirector-AC18 1.17.0 2070

return list

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.

* To get a set containing all elements that are contained at least in one of these collections use [union].

* Returns a set containing all elements that are contained by both this array and the specified collection.

* The returned set preserves the element iteration order of the original array.
element iteration order of the original array.

```kotlin
public infix fun IntArray.subtract(other: Iterable<Int>): Set<Int> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public infix fun LongArray.subtract(other: Iterable<Long>): Set<Long> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public infix fun FloatArray.subtract(other: Iterable<Float>): Set<Float> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public infix fun DoubleArray.subtract(other: Iterable<Double>): Set<Double> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public infix fun BooleanArray.subtract(other: Iterable<Boolean>): Set<Boolean> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public infix fun CharArray.subtract(other: Iterable<Char>): Set<Char> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

public fun <T> Array<out T>.toMutableSet(): MutableSet<T> {
    val set = LinkedHashSet<T>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}
```

The returned set preserves the element iteration order of the original array.

```kotlin
public fun ByteArray.toMutableSet(): MutableSet<Byte> {
    val set = LinkedHashSet<Byte>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun ShortArray.toMutableSet(): MutableSet<Short> {
    val set = LinkedHashSet<Short>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun IntArray.toMutableSet(): MutableSet<Int> {
    val set = LinkedHashSet<Int>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun LongArray.toMutableSet(): MutableSet<Long> {
    val set = LinkedHashSet<Long>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun FloatArray.toMutableSet(): MutableSet<Float> {
    val set = LinkedHashSet<Float>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun DoubleArray.toMutableSet(): MutableSet<Double> {
    val set = LinkedHashSet<Double>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun BooleanArray.toMutableSet(): MutableSet<Boolean> {
    val set = LinkedHashSet<Boolean>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}

public fun CharArray.toMutableSet(): MutableSet<Char> {
    val set = LinkedHashSet<Char>(mapCapacity(size))
    for (item in this)
        set.add(item)
    return set
}
```

Those elements of the [other] collection that are unique are iterated in the
end\n* in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun \(<T> Array\langle<T\rangle.\union\(other: \text{Iterable}\(<T\>)\): \text{Set}\(<T>\) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun ByteArray.\union\(other: \text{Iterable}\(<\text{Byte}\>)\): \text{Set}\(<\text{Byte}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun ShortArray.\union\(other: \text{Iterable}\(<\text{Short}\>)\): \text{Set}\(<\text{Short}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun IntArray.\union\(other: \text{Iterable}\(<\text{Int}\>)\): \text{Set}\(<\text{Int}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun LongArray.\union\(other: \text{Iterable}\(<\text{Long}\>)\): \text{Set}\(<\text{Long}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun DoubleArray.\union\(other: \text{Iterable}\(<\text{Double}\>)\): \text{Set}\(<\text{Double}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun FloatArray.\union\(other: \text{Iterable}\(<\text{Float}\>)\): \text{Set}\(<\text{Float}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun BooleanArray.\union\(other: \text{Iterable}\(<\text{Boolean}\>)\): \text{Set}\(<\text{Boolean}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns a set containing all distinct elements from both collections.\n* \n* The returned set preserves the element iteration order of the original array.\n* Those elements of the [other] collection that are unique are iterated in the end in the order of the [other] collection.\n* \n* To get a set containing all elements that are contained in both collections use \[intersect\].\n* /\npublic infix fun CharArray.\union\(other: \text{Iterable}\(<\text{Char}\>)\): \text{Set}\(<\text{Char}\>) \{\nval set = this.toMutableSet()\nset.addAll(other)\nreturn set\}\n\} /\n\* Returns ’true’ if all elements match the given [predicate].\n* \n* @sample samples.collections.Collections.Aggregates.all.\union\(<\text{T}\rangle Array\langle\text{out}\text{T}\rangle all\(predicate: (T) ->\text{Boolean}\): \text{Boolean} \{\n\} for (element in this) if (\(\text{predicate(element)}\) return false) return true\}\n\} /\n\* Returns ’true’ if all elements match the given [predicate].\n* \n* @sample samples.collections.Collections.Aggregates.all.\union\(<\text{ByteArray}\all\(predicate: (\text{Byte}) ->\text{Boolean}\): \text{Boolean} \{\n\} for (element in this) if (\(\text{predicate(element)}\) return false) return true\} \n\} /\n\* Returns ’true’ if all elements match the given [predicate].\n* \n* @sample samples.collections.Collections.Aggregates.all.\union\(<\text{ShortArray}\all\(predicate: (\text{Short}) ->\text{Boolean}\): \text{Boolean} \{\n\} for (element in this) if (\(\text{predicate(element)}\) return false) return true\} \n\}
(!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun IntArray.all(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun LongArray.all(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun DoubleArray.all(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun BooleanArray.all(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
public inline fun CharArray.all(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun ByteArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun ShortArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun IntArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun LongArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun FloatArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun DoubleArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun BooleanArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if array has at least one element.
 * 
 * @sample samples.collections.Collections.Aggregates.any
 */
public fun CharArray.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun <T> Array<out T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ByteArray.any(predicate: (Byte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun ShortArray.any(predicate: (Short) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun IntArray.any(predicate: (Int) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun LongArray.any(predicate: (Long) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun DoubleArray.any(predicate: (Double) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun BooleanArray.any(predicate: (Boolean) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.anyWithPredicate
 */
public inline fun CharArray.any(predicate: (Char) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}
public inline fun <T> Array<out T>.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun ByteArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun ShortArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun IntArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun LongArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun FloatArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun DoubleArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun BooleanArray.count(): Int {
    return size
}

/**
 * Returns the number of elements in this array.
 */
public inline fun CharArray.count(): Int {
    return size
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun <T> Array<out T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun ByteArray.count(predicate: (Byte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun ShortArray.count(predicate: (Short) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun IntArray.count(predicate: (Int) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun LongArray.count(predicate: (Long) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun FloatArray.count(predicate: (Float) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun DoubleArray.count(predicate: (Double) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun BooleanArray.count(predicate: (Boolean) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given 
 * [predicate].
 */
public inline fun CharArray.count(predicate: (Char) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and
 * applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <T, R> Array<out T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this)
accumulator = operation(accumulator, element)\n    return accumulator

* Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.

*/

public inline fun <R> IntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> ShortArray.fold(initial: R, operation: (index: Int, acc: R, Short) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> BooleanArray.fold(initial: R, operation: (acc: R, Boolean) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> DoubleArray.fold(initial: R, operation: (acc: R, Double) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> FloatArray.fold(initial: R, operation: (acc: R, Float) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> LongArray.fold(initial: R, operation: (acc: R, Long) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> CharArray.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
public inline fun <R> ByteArray.fold(initial: R, operation: (acc: R, Byte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> LongArray.foldIndexed(initial: R, operation: (Int, acc: R, Long) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> FloatArray.foldIndexed(initial: R, operation: (Int, acc: R, Float) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> CharArray.foldIndexed(initial: R, operation: (Int, acc: R, Char) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> IntArray.foldIndexed(initial: R, operation: (Int, acc: R, Int) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> BooleanArray.foldIndexed(initial: R, operation: (Int, acc: R, Boolean) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> DoubleArray.foldIndexed(initial: R, operation: (Int, acc: R, Double) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}

return accumulator
* Accumulates value starting with [initial] value and applying [operation] from left to right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value,\n* and the element itself, and calculates the next accumulator value.\n* /\n
public inline fun <R> ByteArray.foldIndexed(initial: R, operation: (Byte, acc: R) -> R): R {\n\n    var index = lastIndex\n    var accumulator = initial\n    for (element in this) accumulator = operation(index++, accumulator, element)\n    return accumulator
}
FloatArray.foldRight(initial: R, operation: (Float, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> DoubleArray.foldRight(initial: R, operation: (Double, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> LongArray.foldRight(initial: R, operation: (Int, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> IntArray.foldRight(initial: R, operation: (Int, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> ByteArray.foldRight(initial: R, operation: (Byte, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> CharArray.foldRight(initial: R, operation: (Char, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> BooleanArray.foldRight(initial: R, operation: (Boolean, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> ShortArray.foldRight(initial: R, operation: (Short, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}

Accumulates value starting with [initial] value and applying [operation] from right to left
to each element and current accumulator value.

*public inline fun <R> FloatArray.foldRight(initial: R, operation: (Float, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(accumulator, get(index--))
  }
  return accumulator
}
DoubleArray.foldRightIndexed(initial: R, operation: (index: Int, Double, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

BooleanArray.foldRightIndexed(initial: R, operation: (index: Int, Boolean, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

CharArray.foldRightIndexed(initial: R, operation: (index: Int, Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

forEachIndexed
LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

LongArray.forEachIndexed(action: (index: Int, Long) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ByteArray.forEachIndexed(action: (index: Int, Byte) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

DoubleArray.forEachIndexed(action: (index: Int, Double) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

IntArray.forEachIndexed(action: (index: Int, Int) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

ShortArray.forEachIndexed(action: (index: Int, Short) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

CharArray.forEachIndexed(action: (index: Int, Char) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}

FloatArray.forEachIndexed(action: (index: Int, Float) -> Unit): Unit {\n    var index = 0\n    for (item in this) \n        action(index++, item)\n}
1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
ByteArray.maxBy(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
ShortArray.maxBy(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
IntArray.maxBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
LongArray.maxBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
FloatArray.maxBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
DoubleArray.maxBy(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
BooleanArray.maxBy(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}

/**
 * Returns the first element yielding the largest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>>
CharArray.maxBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    }
    return maxElem
}
in 1..lastIndex) { \n    val e = this[i] \n    val v = selector(e) \n    if (maxValue < v) { \n        maxValue = v \\
    } \n    return maxElem
} \n
/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun <T> Array<out T>.maxWith(comparator: Comparator<in T>): T? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun ByteArray.maxWith(comparator: Comparator<in Byte>): Byte? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun ShortArray.maxWith(comparator: Comparator<in Short>): Short? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the smallest
elements.
 */
public fun ShortArray.min(): Short? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (min > e) min = e \\
    } \n    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun <T : Comparable<T>> Array<out T>.min(): T? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun IntArray.min(): Int? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun DoubleArray.maxWith(comparator: Comparator<in Double>): Double? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun ByteArray.maxWith(comparator: Comparator<in Byte>): Byte? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun BooleanArray.maxWith(comparator: Comparator<in Boolean>): Boolean? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun CharArray.maxWith(comparator: Comparator<in Char>): Char? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun FloatArray.maxWith(comparator: Comparator<in Float>): Float? { \n    if (isEmpty()) return null \n    var max = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(max, e) < 0) max = e \\
    } \n    return max
}

/**
 * Returns the smallest
elements.
 */
public fun FloatArray.min(): Float? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (e.isNaN()) return e \n        if (min > e) min = e
        } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun IntArray.min(): Int? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (comparator.compare(min, e) < 0) min = e \\
    } \n    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun ByteArray.min(): Byte? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (min > e) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (min > e) min = e \\
    } \n    return min
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or `null` if there are no elements.
 */
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? { \n    if (isEmpty()) return null \n    var min = this[0] \n    for (i in 1..lastIndex) { \n        val e = this[i] \n        if (min > e) min = e \\
    } \n    return min
}
element or `null` if there are no elements.

```java
/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun IntArray.min(): Int? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun LongArray.min(): Long? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
public fun FloatArray.min(): Float? {
    if (isEmpty()) return null
    var min = this[0]
    if (min.isNaN()) return min
    for (i in 1..lastIndex) {
        val e = this[i]
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun DoubleArray.min(): Double? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public fun CharArray.min(): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <T, R : Comparable<R>> Array<out T>.minBy(selector: (T) -> R): T? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> ByteArray.minBy(selector: (Byte) -> R): Byte? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> ShortArray.minBy(selector: (Short) -> R): Short? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> IntArray.minBy(selector: (Int) -> R): Int? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> LongArray.minBy(selector: (Long) -> R): Long? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> FloatArray.minBy(selector: (Float) -> R): Float? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> CharArray.minBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}
```
1..lastIndex) { val e = this[i] val v = selector(e) if (minValue > v) { minElem = e
minValue = v
} return minElem
}

/**
 * Returns the first element yielding the smallest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> DoubleArray.minBy(selector: (Double) -> R): Double? {
    if (isEmpty()) return null
    var minElem = this[0] val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { val e = this[i] val v = selector(e) if (minValue > v) { minElem = e
minValue = v
} return minElem
}

/**
 * Returns the first element yielding the smallest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> BooleanArray.minBy(selector: (Boolean) -> R): Boolean? {
    if (isEmpty()) return null
    var minElem = this[0] val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { val e = this[i] val v = selector(e) if (minValue > v) { minElem = e
minValue = v
} return minElem
}

/**
 * Returns the first element yielding the smallest value
 * of the given function or `null` if there are no elements.
 */
public inline fun <R : Comparable<R>> CharArray.minBy(selector: (Char) -> R): Char? {
    if (isEmpty()) return null
    var minElem = this[0] val lastIndex = this.lastIndex
    if (lastIndex == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1..lastIndex) { val e = this[i] val v = selector(e) if (minValue > v) { minElem = e
minValue = v
} return minElem
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun <T> Array<out T>.minWith(comparator: Comparator<in T>): T? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun ByteArray.minWith(comparator: Comparator<in Byte>): Byte? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun ShortArray.minWith(comparator: Comparator<in Short>): Short? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun IntArray.minWith(comparator: Comparator<in Int>): Int? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun LongArray.minWith(comparator: Comparator<in Long>): Long? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun FloatArray.minWith(comparator: Comparator<in Float>): Float? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun DoubleArray.minWith(comparator: Comparator<in Double>): Double? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}

/**
 * Returns the first element having the smallest value
 * according to the provided [comparator] or `null` if there are no elements.
 */
public fun BooleanArray.minWith(comparator: Comparator<in Boolean>): Boolean? {
    if (isEmpty()) return null
    var min = this[0] for (i in 1..lastIndex) { val e = this[i] if (comparator.compare(min, e) > 0) min = e
} return min
}
```kotlin
/**
 * Returns `true` if there are no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun <T> Array<out T>.none(): Boolean {
    return isEmpty()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Array<out T>.noneWithPredicate(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun ByteArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun ShortArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun IntArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun LongArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun FloatArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun DoubleArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun BooleanArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if the array has no elements.
 * 
 * @sample samples.collections.Collections.Aggregates.none
 */
public fun CharArray.none(): Boolean {
    return isEmpty()
}
```

```
/**
 * Returns `true` if no elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Array<out T>.noneWithPredicate(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```

```
smallest value according to the provided [comparator] or `null` if there are no elements.
*/
public fun CharArray.minWith(comparator: Comparator<in Char>): Char? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}
```
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

```kotlin
public inline fun <S, T : S> Array<out T>.reduce(operation: (acc: S, T) -> S): S {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

```kotlin
IntArray.reduce(operation: (acc: Int, Int) -> Int): Int {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

```kotlin
ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element.

```kotlin
ByteArray.reduce(operation: (acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```
value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun <S, T : S>
Array<out T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator: S = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
ByteArray.reduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): Byte {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
ShortArray.reduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): Short {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
IntArray.reduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): Int {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
FloatArray.reduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): Float {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to
 * right
 * to current accumulator value and each element with its index in the original array.
 * @param [operation]
 * function that takes the index of an element, current accumulator value
 * and the element itself and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduce
 */
public inline fun
DoubleArray.reduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): Double {
    if (isEmpty()) throw
UnsupportedOperationException("Empty array can't be reduced.")

    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
function that takes the index of an element, current accumulator value \( \text{un} \) and the element itself and calculates the next accumulator value. \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) 

```kotlin
public inline fun BooleanArray.reduceIndexed(operation: (index: Int, acc: Boolean, Boolean) -> Boolean): Boolean {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right \( \text{un} \) to current accumulator value and each element. Returns null if the array is empty. \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) 

```kotlin
public inline fun CharArray.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(index, accumulator, this[index])
    }
    return accumulator
}

```kotlin
/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right to current accumulator value and each element. Returns null if the array is empty. \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) \( \text{un} \) 

```kotlin
public inline fun <S, T : S> Array<out T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    if (isEmpty())
        return null
    var accumulator: S = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

```kotlin
/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right to current accumulator value and each element. Returns null if the array is empty. \( \text{un} \) \( \text{un} \) \( \text{un} \) 

```kotlin
public inline fun ByteArray.reduceOrNull(operation: (acc: Byte, Byte) -> Byte): Byte? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

```kotlin
/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right to current accumulator value and each element. Returns null if the array is empty. \( \text{un} \) 

```kotlin
public inline fun ShortArray.reduceOrNull(operation: (acc: Short, Short) -> Short): Short? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

```kotlin
/**
 * Accumulates value starting with the first element and applying \[operation\] from left to right to current accumulator value and each element. Returns null if the array is empty. \( \text{un} \) 

```kotlin
public inline fun IntArray.reduceOrNull(operation: (acc: Int, Int) -> Int): Int? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex)
```
Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun DoubleArray.reduceOrNull(operation: (acc: Double, Double) -> Double): Double? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun BooleanArray.reduceOrNull(operation: (acc: Boolean, Boolean) -> Boolean): Boolean? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharArray.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {
    if (isEmpty())
        return null
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
```

Accumulates value starting with the last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Array<out T>.reduceRight(operation: (T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun ByteArray.reduceRight(operation: (Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun ShortArray.reduceRight(operation: (Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun IntArray.reduceRight(operation: (Int, acc: Int) -> Int): Int {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun LongArray.reduceRight(operation: (Long, acc: Long) -> Long): Long {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun FloatArray.reduceRight(operation: (Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
```
(Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun DoubleArray.reduceRight(operation: (Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun BooleanArray.reduceRight(operation: (Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun CharArray.reduceRight(operation: (Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun <S, T : S> Array<out T>.reduceRightIndexed(operation: (index: Int, T, acc: S) -> S): S {
    var index = lastIndex
    if (index < 0) throw
    var accumulator: S = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun ByteArray.reduceRightIndexed(operation: (index: Int, Byte, acc: Byte) -> Byte): Byte {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun ShortArray.reduceRightIndexed(operation: (index: Int, Short, acc: Short) -> Short): Short {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current
 * accumulator value.
 */
public inline fun IntArray.reduceRightIndexed(operation: (index: Int, Int, acc: Int) -> Int) {
    var index = lastIndex
    if (index < 0) throw
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
        --index
    }
    return accumulator
}
< 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun FloatArray.reduceRightIndexed(operation: (index: Int, Float, acc: Float) -> Float): Float {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun DoubleArray.reduceRightIndexed(operation: (index: Int, Double, acc: Double) -> Double): Double {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun BooleanArray.reduceRightIndexed(operation: (index: Int, Boolean, acc: Boolean) -> Boolean): Boolean {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element with its index in the original array and current accumulator value.
 * @param [operation] function that takes the index of an element, the element itself and current accumulator value, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.reduceRight
 */
public inline fun CharArray.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
    var index = lastIndex
    if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}
[operation] from right to left to each element and current accumulator value. Returns null if the array is empty.\n
```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T> reduceRightOrNull(operation: (T, acc: T) -> T): T? {
    var index = lastIndex
    if (index < 0) return null
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(accumulator, get(index--))
    }
    return accumulator
}
```

*Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.*
accumulator

* Returns a list containing successive accumulation values generated by applying
  [operation] from left to right to each element and current accumulator value that starts with [initial] value.

* Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the
  previous value in resulting list.

* @param [operation] function that takes current accumulator value and an element, and calculates the
  next accumulator value.

* @sample samples.collections.Collections.Aggregates.scan

**

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <T, R> Array<T>.scan(initial: R, operation: (acc: R, T) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = scanL<T,R>(initial, operation, size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

**

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R> ByteArray.scan(initial: R, operation: (acc: R, Byte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = scanL<Byte,R>(initial, operation, size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

**

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R> ShortArray.scan(initial: R, operation: (acc: R, Short) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = scanL<Short,R>(initial, operation, size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

**

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R> IntArray.scan(initial: R, operation: (acc: R, Int) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = scanL<Int,R>(initial, operation, size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

**

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R> LongArray.scan(initial: R, operation: (acc: R, Long) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = scanL<Long,R>(initial, operation, size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}
[operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.

Note that ‘acc’ value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@since Kotlin("1.3")
@kotlin.experimental.InlineOnly
public inline fun <R> FloatArray.scan(initial: R, operation: (acc: R, Float) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

@since Kotlin("1.3")
@kotlin.experimental.InlineOnly
public inline fun <R> DoubleArray.scan(initial: R, operation: (acc: R, Double) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

@since Kotlin("1.3")
@kotlin.experimental.InlineOnly
public inline fun <R> BooleanArray.scan(initial: R, operation: (acc: R, Boolean) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

@since Kotlin("1.3")
@kotlin.experimental.InlineOnly
public inline fun <R> CharArray.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

@since Kotlin("1.3")
@kotlin.experimental.InlineOnly
public inline fun <T, R> Array<out T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
next accumulator value.

```kotlin
@SinceKotlin("1.3")@ExperimentalStdlibApi
public inline fun <R>
ByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Byte) -> R): List<R> {
    if (isEmpty()) return
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.3")@ExperimentalStdlibApi
public inline fun <R>
ShortArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Short) -> R): List<R> {
    if (isEmpty()) return
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.3")@ExperimentalStdlibApi
public inline fun <R>
IntArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Int) -> R): List<R> {
    if (isEmpty()) return
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.3")@ExperimentalStdlibApi
public inline fun <R>
    if (isEmpty()) return
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

```kotlin
@SinceKotlin("1.3")@ExperimentalStdlibApi
public inline fun <R>
FloatArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Float) -> R): List<R> {
    if (isEmpty()) return
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
otherwise it would affect the previous value in resulting list.

@sinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
DoubleArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Double) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * @param [operation]
 * function that takes the index of an element, current accumulator value and the element itself, and calculates the
 * previous value in resulting list.
 * * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous
 * value in resulting list.
 * @sample samples.collections.Collections.Aggregates.scan
 */

@sinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
BooleanArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Boolean) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * @param [operation]
 * function that takes the index of an element, current accumulator value and the element itself, and calculates the
 * next accumulator value.
 * * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the
 * previous value in resulting list.
 * @sample samples.collections.Collections.Aggregates.scan
 */

@sinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <R>
CharArray.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first element of
 * this array.
 * @param [operation]
 * function that takes current accumulator value and an element, and calculates the next accumulator value.
 * * @sample samples.collections.Collections.Aggregates.scan
 */

@sinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T> Array<out T>.scanReduce(operation: (acc: S, T) -> S): List<S> {
    if (isEmpty()) return emptyList()
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 * @param [operation]
 * function that takes current accumulator value and an element, and calculates the next accumulator value.
 * * @sample samples.collections.Collections.Aggregates.scan
 */

@sinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun ByteArray.scanReduce(operation: (acc: Byte, Byte) -> Byte): List<Byte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 * @param [operation]
 * function that takes current accumulator value and an element, and calculates the next accumulator value.
 * * @sample samples.collections.Collections.Aggregates.scan
 */

@sinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <T> scanReduce(operation: (acc: T, T) -> T): List<T> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<T>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
accumulator.value

@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun ShortArray.scanReduce(operation: (acc: Short, Short) -> Short): List<Short> { \n    var accumulator = this[0] \n    val result = ArrayList<Short>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun IntArray.scanReduce(operation: (acc: Int, Int) -> Int): List<Int> { \n    var accumulator = this[0] \n    val result = ArrayList<Int>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun LongArray.scanReduce(operation: (acc: Long, Long) -> Long): List<Long> { \n    var accumulator = this[0] \n    val result = ArrayList<Long>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun FloatArray.scanReduce(operation: (acc: Float, Float) -> Float): List<Float> { \n    var accumulator = this[0] \n    val result = ArrayList<Float>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun DoubleArray.scanReduce(operation: (acc: Double, Double) -> Double): List<Double> { \n    var accumulator = this[0] \n    val result = ArrayList<Double>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce

*\@SinceKotlin("1.3")\@ExperimentalStdlibApi\@kotlin.internal.InlineOnly\@public inline fun BooleanArray.scanReduce(operation: (acc: Boolean, Boolean) -> Boolean): List<Boolean> { \n    var accumulator = this[0] \n    val result = ArrayList<Boolean>(size).apply { add(accumulator) } \n    for (index in 1 until size) { \n        accumulator = operation(accumulator, this[index]) \n        result.add(accumulator) \n    } \n    return result \n} \n
@sample samples.collections.Collections.Aggregates.scanReduce
and an element, and calculates the next accumulator value.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharArray.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    var accumulator = this[0]
    val result = ArrayList<Char>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first
 * element of this array. Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original array and current accumulator value
 * that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Array<out T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    var accumulator: S = this[0]
    val result = ArrayList<S>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first
 * element of this array. Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original array and current accumulator value
 * that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun ByteArray.scanReduceIndexed(operation: (index: Int, acc: Byte, Byte) -> Byte): List<Byte> {
    var accumulator = this[0]
    val result = ArrayList<Byte>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first
 * element of this array. Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original array and current accumulator value
 * that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun ShortArray.scanReduceIndexed(operation: (index: Int, acc: Short, Short) -> Short): List<Short> {
    var accumulator = this[0]
    val result = ArrayList<Short>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with the first
 * element of this array. Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element, its index in the original array and current accumulator value
 * that starts with the first element of this array.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself,
 * and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun IntArray.scanReduceIndexed(operation: (index: Int, acc: Int, Int) -> Int): List<Int> {
    var accumulator = this[0]
    val result = ArrayList<Int>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```
**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Long>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
FloatArray.scanReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Float>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.scanReduce Indexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Double>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add (accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Char>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Long>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
FloatArray.scanReduceIndexed(operation: (index: Int, acc: Float, Float) -> Float): List<Float> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Float>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
DoubleArray.scanReduceIndexed(operation: (index: Int, acc: Double, Double) -> Double): List<Double> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Double>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
CharArray.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty())
        return emptyList()
    var accumulator = this[0]
    var result = ArrayList<Char>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
```

**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
CharArray.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**@SinceKotlin("1.3")**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ByteArray.sumBy(selector: (Byte) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```

**Since Kotlin 1.3**

```kotlin
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
ShortArray.sumBy(selector: (Short) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```
Returns the sum of all values produced by [selector] function applied to each element in the array.

inline fun IntArray.sumBy(selector: (Int) -> Int): Int {\n    var sum: Int = 0
    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}*\n
Returns the sum of all values produced by [selector] function applied to each element in the array.

public fun <T> Array<out T>.sumByDouble(selector: (T) -> Double): Double {\n    var sum: Double = 0.0
    for (element in this) {\n        sum += selector(element)\n    }\n    return sum\n}*\n
Returns the sum of all values produced by [selector] function applied to each element in the array.

public inline fun <T> Array<out T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {\n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {\n        if (predicate(element)) {\n            first.add(element)\n        } else {\n            second.add(element)\n        }\n    }\n    return first to second\n}*\n
Please note that the code snippet contains incomplete annotations and some unbalanced braces, which may affect the correctness of the generated code.
second.add(element)
    } else {
        second.add(element)
    }
}

fun ByteArray.partition(predicate: (Byte) -> Boolean): Pair<List<Byte>, List<Byte>> {
    val first = ArrayList<Byte>()
    val second = ArrayList<Byte>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

fun IntArray.partition(predicate: (Int) -> Boolean): Pair<List<Int>, List<Int>> {
    val first = ArrayList<Int>()
    val second = ArrayList<Int>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

fun ShortArray.partition(predicate: (Short) -> Boolean): Pair<List<Short>, List<Short>> {
    val first = ArrayList<Short>()
    val second = ArrayList<Short>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

fun CharArray.partition(predicate: (Char) -> Boolean): Pair<List<Char>, List<Char>> {
    val first = ArrayList<Char>()
    val second = ArrayList<Char>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}
** Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.

* The returned list has length of the shortest collection.

@sample samples.collections.Operations.zipIterable

```kotlin
fun <R> ShortArray.zip(other: Array<out R>): List<Pair<Short, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> IntArray.zip(other: Array<out R>): List<Pair<Int, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> LongArray.zip(other: Array<out R>): List<Pair<Long, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> FloatArray.zip(other: Array<out R>): List<Pair<Float, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> DoubleArray.zip(other: Array<out R>): List<Pair<Double, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> BooleanArray.zip(other: Array<out R>): List<Pair<Boolean, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <R> CharArray.zip(other: Array<out R>): List<Pair<Char, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
```

```kotlin
fun <T, R, V> Array<out T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> ByteArray.zip(other: Array<out R>, transform: (a: Byte, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> ShortArray.zip(other: Array<out R>, transform: (a: Short, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> IntArray.zip(other: Array<out R>, transform: (a: Int, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> LongArray.zip(other: Array<out R>, transform: (a: Long, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> FloatArray.zip(other: Array<out R>, transform: (a: Float, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> DoubleArray.zip(other: Array<out R>, transform: (a: Double, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> BooleanArray.zip(other: Array<out R>, transform: (a: Boolean, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <R, V> CharArray.zip(other: Array<out R>, transform: (a: Char, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```

```kotlin
fun <T, R, V> Array<out T>.zip(other: Array<out R>, transform: (a: T, b: R) -> V): List<V> {
    val list = ArrayList<V>(other.size)
    for (i in 0 until other.size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}
```
* using the provided [transform] function applied to each pair of elements. The returned list has length of the shortest collection.

* @sample 

*之星

* Returns a list of values built from the elements of ‘this’ array and the [other] array with the same index

* using the provided [transform] function applied to each pair of elements. The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample 

* The returned list has length of the shortest collection.

* @sample
FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>>

DoubleArray.zip(other: Iterable<R>): List<Pair<Double, R>>

BooleanArray.zip(other: Iterable<R>): List<Pair<Boolean, R>>

CharArray.zip(other: Iterable<R>): List<Pair<Char, R>>

ByteArray.zip(other: Iterable<R>): List<V>

ShortArray.zip(other: Iterable<R>): List<V>

IntArray.zip(other: Iterable<R>): List<V>

LongArray.zip(other: Iterable<R>): List<V>

FloatArray.zip(other: Iterable<R>): List<Pair<Float, R>>

```kotlin
FloatArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

DoubleArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

BooleanArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

CharArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

ByteArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

ShortArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

IntArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

LongArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }

FloatArray.zip(other: Iterable<R>) { t1, t2 -> t1 to t2 }
```
ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> DoubleArray.zip(other: Iterable<R>, transform: (a: Double, b: R) -> V): List<V> {
    val arraySize = size
    val list = ArrayList<V>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> BooleanArray.zip(other: Iterable<R>, transform: (a: Boolean, b: R) -> V): List<Pair<Boolean, V>> {
    val arraySize = size
    val list = ArrayList<Pair<Boolean, V>>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> CharArray.zip(other: Iterable<R>, transform: (a: Char, b: R) -> V): List<Pair<Char, V>> {
    val arraySize = size
    val list = ArrayList<Pair<Char, V>>(minOf(other.collectionSizeOrDefault(10), arraySize))
    var i = 0
    for (element in other) {
        if (i >= arraySize) break
        list.add(transform(this[i++], element))
    }
    return list
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> ByteArray.zip(other: ByteArray, transform: (a: Byte, b: R) -> V): List<Pair<Byte, V>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> ShortArray.zip(other: ShortArray, transform: (a: Short, b: R) -> V): List<Pair<Short, V>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> IntArray.zip(other: IntArray, transform: (a: Int, b: R) -> V): List<Pair<Int, V>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> FloatArray.zip(other: FloatArray, transform: (a: Float, b: R) -> V): List<Pair<Float, V>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index
 * using the provided [transform] function applied to each pair of elements.
 * The returned list has length of the shortest collection.
 */
fun <R, V> DoubleArray.zip(other: DoubleArray, transform: (a: Double, b: R) -> V): List<Pair<Double, V>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
given [prefix] and [postfix] if supplied.

* If the collection could be huge, you can specify a non-negative value of 
  [limit], in which case only the first [limit] * elements will be appended, 
  followed by the [truncated] string (which defaults to "...").

* @sample samples.collections.Collections.Transformations.joinTo

```kotlin
public fun <A : Appendable> ByteArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", 
    postfix: CharSequence = ", ", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = 
    null): A {
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    } else break

    buffer.append(separator)
    if (limit < 0 || count <= limit) {
        if (transform != null)
            buffer.appendElement(element, transform)
    } else break

    buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break

    buffer.append(separator)
    var count = 0
    for (element in this) {
        if (++count > 1)
            buffer.append(transform(element))
        else
            buffer.append(element.toString())
    } else break
```
buffer

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> CharArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): A {
    return
}

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> BooleanArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Boolean) -> CharSequence)? = null): A {
    return
}

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> IntArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Int) -> CharSequence)? = null): A {
    return
}

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> LongArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Long) -> CharSequence)? = null): A {
    return
}

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> DoubleArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Double) -> CharSequence)? = null): A {
    return
}

Appends the string from all the elements separated using [separator] and using the given
[prefix] and [postfix] if supplied. If the collection could be huge, you can specify a non-negative value of
[limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which
defaults to "..."), @sample samples.collections.Collections.Transformations.joinTo

public fun <A : Appendable> FloatArray.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Float) -> CharSequence)? = null): A {
    return
}
If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.

If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").

If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").

String | return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform), toString()) | n | */

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

public fun CharArray.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((Char) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform), toString()) | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun ByteArray.asIterable(): Iterable<Byte> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun ShortArray.asIterable(): Iterable<Short> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun IntArray.asIterable(): Iterable<Int> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun LongArray.asIterable(): Iterable<Long> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun FloatArray.asIterable(): Iterable<Float> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun DoubleArray.asIterable(): Iterable<Double> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.
 */

public fun BooleanArray.asIterable(): Iterable<Boolean> {
    if (isEmpty()) return emptyList()
    return Iterable { this.iterator() | n | /*

Creates an [Iterable] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */

public fun ByteArray.asSequence(): Sequence<Byte> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() | n | /*

Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */

public fun ShortArray.asSequence(): Sequence<Short> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() | n | /*

Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */

public fun IntArray.asSequence(): Sequence<Int> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() | n | /*

Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.

*/

/**
 * Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.
 */

public fun LongArray.asSequence(): Sequence<Long> {
    if (isEmpty()) return emptySequence()
    return Sequence { this.iterator() | n | /*

Opensourceusedin IOx-FogDirector-AC18 1.17.0 2109
Creates a [Sequence] instance that wraps the original array returning its elements when being iterated.

```kotlin
*\n Ln* Ln* @sample samples.collections.Sequences.Building.sequenceFromArray* \n *\n inpUBLIC fun BooleanArray.asSequence(): Sequence<Boolean> {\n    if (isEmpty()) return emptySequence() return Sequence {\n        this.iterator() }\n    }\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } return if (count == 0) Double.NaN\n    else sum / count\n}

Ln* Ln* @sample samples.collections.Sequences.Building.sequenceFromArray* \n *\n inpUBLIC fun CharArray.asSequence(): Sequence<Char> {\n    if (isEmpty()) return emptySequence() return Sequence {\n        this.iterator() }\n    }\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } return if (count == 0) Double.NaN\n    else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfByte") \n *\n inpUBLIC fun Array<out Byte>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfShort") \n *\n inpUBLIC fun Array<out Short>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfInt") \n *\n inpUBLIC fun Array<out Int>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfLong") \n *\n inpUBLIC fun Array<out Long>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfFloat") \n *\n inpUBLIC fun Array<out Float>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfDouble") \n *\n inpUBLIC fun Array<out Double>.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfByte") \n *\n inpUBLIC fun ByteArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfShort") \n *\n inpUBLIC fun ShortArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfInt") \n *\n inpUBLIC fun IntArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfLong") \n *\n inpUBLIC fun LongArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfFloat") \n *\n inpUBLIC fun FloatArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("averageOfDouble") \n *\n inpUBLIC fun DoubleArray.average(): Double {\n    var sum: Double = 0.0\n    var count: Int = 0\n    for (element in this) \n        sum += element\n    } if (count == 0) \n        return Double.NaN else sum / count\n}

Ln* Ln* @kotlin.jvm.JvmName("sumOfByte") \n *\n inpUBLIC fun Array<out Byte>.sum(): Int {\n    var sum: Int = 0\n    for (element in this) \n        sum += element\n    } return sum\n}

Ln* Ln* @kotlin.jvm.JvmName("sumOfShort") \n *\n inpUBLIC fun Array<out Short>.sum(): Int {\n    var sum: Int = 0\n    for (element in this) \n        sum += element\n    } return sum\n}

Ln* Ln* @kotlin.jvm.JvmName("sumOfInt") \n *\n inpUBLIC fun Array<out Int>.sum(): Int {\n    var sum: Int = 0\n    for (element in this) \n        sum += element\n    } return sum\n}

Ln* Ln* @kotlin.jvm.JvmName("sumOfLong") \n *\n inpUBLIC fun Array<out Long>.sum(): Long {\n    var sum: Long = 0\n    for (element in this) \n        sum += element\n    } return sum\n}
```
Long = 0L
for (element in this) {
    sum += element
}
return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun Array<out Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun ByteArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun ShortArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun IntArray.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun LongArray.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun FloatArray.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the array.
 */
public fun DoubleArray.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun IntRange.random(): Int {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun LongRange.random(): Long {
    return random(Random)
}

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun CharRange.random(): Char {
    return random.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun IntRange.randomOrNull(): Int? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun LongRange.randomOrNull(): Long? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun
CharRange.randomOrNull(): Char? {
    return randomOrNull(Random())
} * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.

findFirst(): Int? {
    return randomOrNull(Random())
}

/**
 * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun IntRange.randomOrNull(random: Random): Int? {
    if (isEmpty())
        return randomOrNull.nextInt(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun LongRange.randomOrNull(random: Random): Long? {
    if (isEmpty())
        return LongRange.randomOrNull.nextLong(this)
}

/**
 * Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun CharRange.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return CharRange.randomOrNull.nextInt(first.toInt(), last.toInt() + 1).toChar()
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Int?): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Long?): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun ClosedRange<Byte>.contains(value: Byte): Boolean {
    return contains(value.toByte())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("shortRangeContains")
public operator fun ClosedRange<Short>.contains(value: Short): Boolean {
    return contains(value.toShort())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun ClosedRange<Double>.contains(value: Double): Boolean {
    return contains(value.toDouble())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun ClosedRange<Float>.contains(value: Float): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("intRangeContains")
public operator fun ClosedRange<Int>.contains(value: Double): Boolean {
    return value.toIntExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@SinceKotlin("1.3")
@kotlin.jvm.JvmName("longRangeContains")
public operator fun ClosedRange<Long>.contains(value: Short): Boolean {
    return value.toLongExactOrNull().let { if (it != null)
        contains(it) else false }
}

*/
ClosedRange<Short>.contains(value: Double): Boolean { 
    return value.toShortExactOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Float>.contains(value: Double): Boolean { 
    return value.toFloatOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Float): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Float): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Long): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Long): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Float>.contains(value: Long): Boolean { 
    return value.toFloatOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Long): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Long): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Double): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Double): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Double): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Double): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Float): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Float): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Float): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Float): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Long): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Long): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Long): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Long): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Double): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Double): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Double): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Double): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Float): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Float): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Float): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Float): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Long): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Long): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Long): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Long): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Double): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Double): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Double): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Double): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Float): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Float): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Float): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Long>.contains(value: Float): Boolean { 
    return value.toLongOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Byte>.contains(value: Long): Boolean { 
    return value.toByteOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Short>.contains(value: Long): Boolean { 
    return value.toShortOrNull().let { if (it != null) 
        contains(it) else false }
}

ClosedRange<Int>.contains(value: Long): Boolean { 
    return value.toIntOrNull().let { if (it != null) 
        contains(it) else false }
}
Checks if the specified [value] belongs to this range.

```kotlin
Checks if the specified [value] belongs to this range.

// @Deprecated("This `contains` operation mixing integer and floating point arguments has ambiguous semantics and is going to be removed.")
@kotlin.jvm.JvmName("floatRangeContains")
public operator fun
ClosedRange<Float>.contains(value: Long): Boolean {
    return contains(value.toFloat())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("intRangeContains")
public operator fun
ClosedRange<Int>.contains(value: Short): Boolean {
    return contains(value.toInt())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("longRangeContains")
public operator fun
ClosedRange<Long>.contains(value: Short): Boolean {
    return contains(value.toLong())
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("byteRangeContains")
public operator fun
ClosedRange<Byte>.contains(value: Short): Boolean {
    return value.toByteExactOrNull().let { if (it != null) contains(it) else false }
}

/**
 * Checks if the specified [value] belongs to this range.
 */
@kotlin.jvm.JvmName("doubleRangeContains")
public operator fun
ClosedRange<Double>.contains(value: Short): Boolean {
    return contains(value.toDouble())
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Int.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Long.downTo(to: Byte): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Byte.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Short.downTo(to: Byte): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Char.downTo(to: Char): CharProgression {
    return CharProgression.fromClosedRange(this, to, -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Int.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this, to, -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Long.downTo(to: Int): LongProgression {
    return LongProgression.fromClosedRange(this, to.toLong(), -1L)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Byte.downTo(to: Int): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to, -1)
}

/**
 * Returns a progression from this value down to the specified [to] value with the step -1.
 * The [to] value should be less than or equal to `this` value.
 * The returned progression is empty.
 */
public infix fun Short.downTo(to: Int): IntProgression {
    return
```
IntProgression.fromClosedRange(this.toInt(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Int.downTo(to: Long): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Int.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this, to.toInt(), -1)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Byte.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Byte.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Short.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Short.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun IntProgression.reversed(): IntProgression {
    return IntProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun LongProgression.reversed(): LongProgression {
    return LongProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun CharProgression.reversed(): CharProgression {
    return CharProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range with the given step.\n\npublic infix fun IntProgression.step(step: Int): IntProgression {
    checkStepIsPositive(step > 0, step)\n    return IntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n\n// Returns a progression that goes over the same range with the given step.\n\npublic infix fun LongProgression.step(step: Long): LongProgression {
    checkStepIsPositive(step > 0, step)\n    return LongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n\n// Returns a progression that goes over the same range with the given step.\n\npublic infix fun CharProgression.step(step: Int): CharProgression {
    checkStepIsPositive(step > 0, step)\n    return CharProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)\n\n// Returns a progression that goes over the same range with the given step.\n\npublic infix fun Byte.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Byte.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Short.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression from this value down to the specified [to] value with the step -1.\n\npublic infix fun Short.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun IntProgression.reversed(): IntProgression {
    return IntProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun LongProgression.reversed(): LongProgression {
    return LongProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun CharProgression.reversed(): CharProgression {
    return CharProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun IntProgression.reversed(): IntProgression {
    return IntProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun LongProgression.reversed(): LongProgression {
    return LongProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun CharProgression.reversed(): CharProgression {
    return CharProgression.fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Short): IntProgression {
    return IntProgression.fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Long): LongProgression {
    return LongProgression.fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Short): IntProgression {
    return IntProgression fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun IntProgression.reversed(): IntProgression {
    return IntProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun LongProgression.reversed(): LongProgression {
    return LongProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun CharProgression.reversed(): CharProgression {
    return CharProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Long): LongProgression {
    return LongProgression fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Short): IntProgression {
    return IntProgression fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Long): LongProgression {
    return LongProgression fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Short): IntProgression {
    return IntProgression fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun IntProgression.reversed(): IntProgression {
    return IntProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun LongProgression.reversed(): LongProgression {
    return LongProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun CharProgression.reversed(): CharProgression {
    return CharProgression fromClosedRange(last, first, -step)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Long): LongProgression {
    return LongProgression fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Byte.downTo(to: Short): IntProgression {
    return IntProgression fromClosedRange(this.toInt(), to.toInt(), -1)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Long): LongProgression {
    return LongProgression fromClosedRange(this.toLong(), to, -1L)\n\n// Returns a progression that goes over the same range in the opposite direction with the same step.\n\npublic fun Short.downTo(to: Short): IntProgression {
    return IntProgression fromClosedRange(this.toInt(), to.toInt(), -1)
Double.toByteExactOrNull(): Byte? { return if (this in Byte.MIN_VALUE.toDouble()..Byte.MAX_VALUE.toDouble()) this.toByte() else null }

internal fun Float.toByteExactOrNull(): Byte? {
    return if (this in Byte.MIN_VALUE.toFloat()..Byte.MAX_VALUE.toFloat()) this.toByte() else null
}

internal fun Long.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toLong()..Int.MAX_VALUE.toLong()) this.toInt() else null
}

internal fun Double.toIntExactOrNull(): Int? {
    return if (this in Int.MIN_VALUE.toDouble()..Int.MAX_VALUE.toDouble()) this.toInt() else null
}

internal fun Byte.toShortExactOrNull(): Short? {
    return if (this in Short.MIN_VALUE.toInt()..Short.MAX_VALUE.toInt()) this.toShort() else null
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Byte): IntRange {
    return this .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Byte): LongRange {
    return this .. (to.toLong() - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Int): IntRange {
    return this .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Int): IntRange {
    return this .. (to - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Char.until(to: Char): CharRange {
    return CharRange.EMPTY
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Long): LongRange {
    return IntRange.EMPTY
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Int): LongRange {
    return LongRange.EMPTY
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Long): LongRange {
    return LongRange.EMPTY
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Long): LongRange {
    return LongRange.EMPTY
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Double): DoubleRange {
    return this .. (to - 1.0).toDouble()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Double): DoubleRange {
    return this .. (to.toLong() - 1.0).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Double): DoubleRange {
    return this .. (to.toDouble() - 1.0).toDouble()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Double): DoubleRange {
    return this .. (to.toDouble() - 1.0).toDouble()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Float): FloatRange {
    return this .. (to.toLong() - 1.0).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Long.until(to: Float): FloatRange {
    return this .. (to.toLong() - 1.0).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Byte.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Short.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 */
public infix fun Int.until(to: Float): FloatRange {
    return this .. (to.toFloat() - 1.0).toFloat()
Long): LongRange {\n if (to <= Long.MIN_VALUE) return LongRange.EMPTY
 return this .. (to - 1).toLong();
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to 'this' value, then the returned range is empty.
 * 
 * @param to the value to excluded
 * @return the range
 */
public infix fun Byte.until(to: Long): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to 'this' value, then the returned range is empty.
 * 
 * @param to the value to excluded
 * @return the range
 */
public infix fun Short.until(to: Long): LongRange {
    if (to <= Long.MIN_VALUE) return LongRange.EMPTY
    return this.toLong() .. (to - 1).toLong()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to 'this' value, then the returned range is empty.
 * 
 * @param to the value to excluded
 * @return the range
 */
public infix fun Int.until(to: Short): IntRange {
    return this .. (to.toInt() - 1).toInt()
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * 
 * If the [to] value is less than or equal to 'this' value, then the returned range is empty.
 * 
 * @param to the value to excluded
 * @return the range
 */
public infix fun Long.until(to: Int): LongRange {
    return this .. (to.toLong() - 1).toLong()
}
than or equal to the [maximumValue] or the [maximumValue] otherwise.

```kotlin
public fun Byte.coerceAtMost(maximumValue: Byte): Byte {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
public fun Short.coerceAtMost(maximumValue: Short): Short {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
public fun Int.coerceAtMost(maximumValue: Int): Int {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
public fun Long.coerceAtMost(maximumValue: Long): Long {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
public fun Float.coerceAtMost(maximumValue: Float): Float {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
public fun Double.coerceAtMost(maximumValue: Double): Double {
    return if (this > maximumValue) maximumValue else this
}
```

```kotlin
/public fun T.coerceInComparable(minimumValue: T?, maximumValue: T?): T {
    if (minimumValue !== null && maximumValue !== null) {
        if (minimumValue > maximumValue) throw
            IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
            minimum $minimumValue.")
        if (this < minimumValue) return minimumValue
        if (this > maximumValue) return maximumValue
    } else {
        if (minimumValue !== null && this < minimumValue) return minimumValue
        if (maximumValue !== null && this > maximumValue) return maximumValue
    }
    return this
}
```

```kotlin
/public fun Byte.coerceIn(minimumValue: Byte, maximumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
    minimum $minimumValue.")
}
```

```kotlin
/public fun Byte.coerceIn(minimumValue: Byte): Byte {
    if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
    minimum $minimumValue.")
}
```

```kotlin
/public fun INT.coerceIn(minimumValue: INT, maximumValue: INT): INT {
    if (minimumValue > maximumValue) throw
    IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than
    minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue)
```
samples.comparisons.ComparableOps.coerceIn

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it’s in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * 
 * @sample samples.comparisons.ComparableOps.coerceIn
 */
public fun <T : Comparable<T>> T.coerceIn(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it’s in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInFloatingPointRange
 */
@SinceKotlin("1.1")
public fun <T : Comparable<T>> T.coerceInFloatingPointRange(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < start equiv to this <= start && !(this >= start) &&
        !range.lessThanOrEquals(start, this) -> range.start
        this > end equiv to this >= end && !(this <= end) &&
        !range.lessThanOrEquals(end, this) -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it’s in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInComparable
 */
public fun <T : Comparable<T>> T.coerceInComparable(range: ClosedRange<T>): T {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<T>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * 
 * @return this value if it’s in the range, or `range.start` if this value is less than `range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive`.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInComparable
 */
@SinceKotlin("1.1")
public fun <T : Comparable<T>> T.coerceInComparableFloatingPointRange(range: ClosedFloatingPointRange<T>): T {
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}
`range.start`, or `range.endInclusive` if this value is greater than `range.endInclusive

if (range is ClosedFloatingPointRange) { return this.coerceIn(range: ClosedRange<Long>): Long

when (this < range.start || this > range.endInclusive) return this

if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")

return when {
    this < range.start -> range.start
    this > range.endInclusive -> range.endInclusive
    else ->
        this
    }

*/

* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.

Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/// Auto-generated file. DO NOT EDIT

package kotlin

import kotlin.experimental.*

@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline class UByte @PublishedApi internal constructor(@PublishedApi internal val data: Byte) : Comparable<UByte> {

    companion object {
        /**
         * A constant holding the minimum value an instance of UByte can have.
         */
        public const val MIN_VALUE: UByte = UByte(0)

        /**
         * A constant holding the maximum value an instance of UByte can have.
         */
        public const val MAX_VALUE: UByte = UByte(-1)

        /**
         * The number of bytes used to represent an instance of UByte in a binary form.
         */
        public const val SIZE_BYTES: Int = 1

        /**
         * The number of bits used to represent an instance of UByte in a binary form.
         */
        public const val SIZE_BITS: Int = 8

        /**
         * Compares this value with the specified value for order.
         * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
         */
        public override inline operator fun compareTo(other: UByte): Int = this.toInt().compareTo(other.toInt())

        /**
         * Compares this value with the specified value for order.
         * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
         */
        public inline operator fun compareTo(other: UShort): Int = this.toInt().compareTo(other.toInt())

        /**
         * Compares this value with the specified value for order.
         * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
         */
        public inline operator fun compareTo(other: UInt): Int = this.toUInt().compareTo(other)

        /**
         * Compares this value with the specified value for order.
         * Returns zero if this value is equal to the specified other value, a negative number if it's less than other, or a positive number if it's greater than other.
         */
        public inline operator fun compareTo(other: ULong): Int = this.toULong().compareTo(other)

        /** Adds the other value to this value. */
        public inline operator fun plus(other: UByte): UInt = this.toUInt().plus(other.toUInt())

        /** Adds the other value to this value. */
        public inline operator fun plus(other: UShort): UInt = this.toUInt().plus(other.toUInt())

        /** Adds the other value to this value. */
        public inline operator fun plus(other: UInt): UInt = this.toUInt().plus(other)

        /** Adds the other value to this value. */
        public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other)

        /** Subtracts the other value from this value. */
        public inline operator fun minus(other: UByte): UInt = this.toUInt().minus(other.toUInt())

        /** Subtracts the other value from this value. */
        public inline operator fun minus(other: UShort): UInt = this.toUInt().minus(other.toUInt())

        /** Subtracts the other value from this value. */
        public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other.toUInt())

        /** Subtracts the other value from this value. */
        public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)

        /** Multiplies this value by the other value. */
        public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())

        /** Multiplies this value by the other value. */
        public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())

        /** Multiplies this value by the other value. */
        public inline operator fun times(other: UInt): UInt = this.toUInt().times(other.toUInt())

        /** Multiplies this value by the other value. */
        public inline operator fun times(other: ULong): ULong = this.toULong().times(other)

        /** Divides this value by the other value. */
        public inline operator fun div(other: UByte): UInt =
@kotlin.internal.InlineOnly
public inline fun toUInt(): UInt = UInt(data.toInt() and 0xFF)

The resulting 'UInt' value represents the same numerical value as this 'UByte'. The least significant 8 bits of the resulting 'UInt' value are the same as the bits of this 'UByte' value whereas the most significant 56 bits are filled with zeros.

Converts this [UByte] value to [ULong].

The resulting 'ULong' value represents the same numerical value as this 'UByte':

![Image](Open Source Used In IOx-FogDirector-AC18 1.17.0 2122)

Converts this [UByte] value to [Float].

The resulting 'Float' value represents the same numerical value as this 'UByte':

![Image](Open Source Used In IOx-FogDirector-AC18 1.17.0 2122)

Converts this [UByte] value to [Double]

The resulting 'Double' value represents the same numerical value as this 'UByte'.

![Image](Open Source Used In IOx-FogDirector-AC18 1.17.0 2122)

Converts this [UByte] value to [Byte]

The resulting 'Byte' value represents the same binary representation as this 'UByte'.

Converts this [UByte] value to [Short]

The resulting 'Short' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Int]

The resulting 'Int' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Long]

The resulting 'Long' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Byte]

The resulting 'Byte' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Short]

The resulting 'Short' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Int]

The resulting 'Int' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Long]

The resulting 'Long' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Byte]

The resulting 'Byte' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Short]

The resulting 'Short' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Int]

The resulting 'Int' value represents the same numerical value as this 'UByte'.

Converts this [UByte] value to [Long]

The resulting 'Long' value represents the same numerical value as this 'UByte'.

The `compareTo()` method can be used to compare two `UByte` values.

A constant holding the minimum value an instance of UInt can have:

A constant holding the maximum value an instance of UInt can have:

The number of bytes used to represent an instance of UInt in a binary form:

The number of bits used to represent an instance of UInt in a binary form:

The specified value of order.

Returns zero if this value is equal to the specified other value, a negative number if it’s less than other or a positive number if it’s greater than other.

Compare this value with the specified value for order.

Returns zero if this value is equal to the specified other value, a negative number if it’s less than other or a positive number if it’s greater than other.

Compare this value with the specified value for order.

Returns zero if this value is equal to the specified other value, a negative number if it’s less than other or a positive number if it’s greater than other.

Compare this value with the specified value for order.

Returns zero if this value is equal to the specified other value, a negative number if it’s less than other or a positive number if it’s greater than other.

Compare this value with the specified value for order.

Returns zero if this value is equal to the specified other value, a negative number if it’s less than other or a positive number if it’s greater than other.
@kotlin.internal.InlineOnly
    @Suppress("OVERRIDE_BY_INLINE")
    public override inline operator fun compareTo(other: UInt): Int = uintCompare(this.data, other.data) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UByte): UInt = this.plus(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UShort): UInt = this.plus(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: UInt): UInt = UInt(this.data.plus(other.data)) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun plus(other: ULong): ULong = this.toULong().plus(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UByte): UInt = this.minus(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UShort): UInt = this.minus(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: UInt): UInt = UInt(this.data.minus(other.data)) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UByte): UInt = this.times(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UShort): UInt = this.times(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: UInt): UInt = UInt(this.data.times(other.data)) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun times(other: ULong): ULong = this.toULong().times(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UByte): UInt = this.div(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UShort): UInt = this.div(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: UInt): UInt = uintDivide(this, other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun div(other: ULong): ULong = this.toULong().div(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UByte): UInt = this.rem(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UShort): UInt = this.rem(other.toUInt()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): UInt = uintRemainder(this, other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): UInt = UInt(data.inc()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): UInt = UInt(data.dec()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: UInt): UInt = UInt(this.data.rem(other.data)) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun inc(): UInt = UInt(data.inc()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun dec(): UInt = UInt(data.dec()) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun rangeTo(other: UInt): UIntRange = UIntRange(this, other) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun shl(bitCount: Int): UInt = UInt(data shl bitCount) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun shr(bitCount: Int): UInt = UInt(data ushr bitCount) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun and(other: UInt): UInt = UInt(this.data and other.data) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun or(other: UInt): UInt = UInt(this.data or other.data) /* Adds the other value to this value. */
    @kotlin.internal.InlineOnly
    public inline operator fun xor(other: UInt): UInt = UInt(this.data xor other.data) /* Adds the other value to this value. */
public inline infix fun xor(other: UInt): UInt = UInt(this.data xor other.data) /* Inverts the bits in this value. */

@kotlin.internal.InlineOnly
public inline fun inv(): UInt = UInt(data.inv()) /* Converts this `UInt` value to [Byte]. */

* If this value is less than or equals to [Byte.MAX_VALUE], the resulting `Byte` value represents the same numerical value as this `UInt`. * The resulting `Byte` value is represented by the least significant 8 bits of this `UInt` value. * Note that the resulting `Byte` value may be negative.

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte() /* Converts this [Byte] value to [Short]. */

* If this value is less than or equals to [Short.MAX_VALUE], the resulting `Short` value represents the same numerical value as this `UInt`. * The resulting `Short` value is represented by the least significant 16 bits of this `UInt` value. * Note that the resulting `Short` value may be negative.

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort() /* Converts this [Short] value to [Int]. */

* If this value is less than or equals to [Int.MAX_VALUE], the resulting `Int` value represents the same numerical value as this `UInt`. Otherwise the result is negative.

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data /* Converts this [Int] value to [Long]. */

* The resulting `Long` value represents the same numerical value as this `UInt`. * The least significant 32 bits of the resulting `Long` value are the same as the bits of this `UInt` value. whereas the most significant 32 bits are filled with zeros.

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong() and 0xFFFFFFFF /* Converts this [UInt] value to [ULong]. */

* If this value is less than or equals to [ULong.MAX_VALUE], the resulting `ULong` value represents the same numerical value as this `UInt`. * The resulting `ULong` value is represented by the least significant 32 bits of this `UInt` value. * In case when this `UInt` value is exactly between two `Float`'s, the one with zero at least significant bit of mantissa is selected.

@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toDouble().toFloat() /* Converts this [UInt] value to [Float]. */

* The resulting `Float` value is represented by the least significant 32 bits of this `UInt` value. * The most significant 32 bits are filled with zeros. whereas the most significant 32 bits are filled with zeros.

@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = this.toDouble().toDouble() /* Converts this [Double] value to [ULong]. */

* The resulting `Double` value represents the same numerical value as this `UInt`. * The least significant 8 bits of the resulting `UInt` value are the same as the bits of this `Byte` value. whereas the most significant 24 bits are filled with the sign bit of this value.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Byte.toUInt(): UInt = UInt(data.toUInt()) /* Converts this [Byte] value to [Int]. */

* If this value is positive, the resulting `UInt` value represents the same numerical value as this `Byte`. * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Byte` value. whereas the most significant 16 bits are filled with the sign bit of this value.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun Short.toUInt(): UInt = UInt(data.toUInt()) /* Converts this [Short] value to [Int]. */

* If this value is positive, the resulting `UInt` value represents the same numerical value as this `Short`. * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `Short` value. whereas the most significant 16 bits are filled with the sign bit of this value.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun
Int.toHexString() = Int.this.toLong().toString(16) * Converts this [Int] value to [Long].

*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly
public inline fun Int.toHexString(): String = Int.toHexString(16)

*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly
public inline fun Int.toHexString(radix: Int): String

/**
 * Converts this [Long] value to [UInt].
 *
 * If this value is positive and less than or equals to [UInt.MAX_VALUE], the resulting `UInt` value represents the same numerical value as this `Long`.
 * The resulting `UInt` value is represented by the least significant 32 bits of this `Long` value.
 *
 * Example:
 * ```
 * println(Int.toHexString(0L)) // 0
 * println(Int.toHexString(1L)) // 1
 * println(Int.toHexString(-1L)) // 4294967295
 * ```
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toHexString(): String = Long.toHexString(16)

/**
 * Converts this [Float] value to [UInt].
 *
 * The fractional part, if any, is rounded down towards zero.
 * Returns zero if this `Double` value is negative or `NaN`, [UInt.MAX_VALUE] if it's bigger than `UInt.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toHexString(): String = Float.toHexString()
@kotlin.internal.InlineOnly
public inline operator fun minus(other: UInt): UInt = this.toUInt().minus(other)
/** Subtracts the other value from this value. */

@kotlin.internal.InlineOnly
public inline operator fun minus(other: ULong): ULong = this.toULong().minus(other)
/** Multiplies this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun times(other: UByte): UInt = this.toUInt().times(other.toUInt())
/** Multiplies this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun times(other: UShort): UInt = this.toUInt().times(other.toUInt())
/** Multiplies this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun times(other: UInt): UInt = this.toUInt().times(other)
/** Multiplies this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun times(other: ULong): ULong = this.toULong().times(other)
/** Divides this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun div(other: UByte): UInt = this.toUInt().div(other.toUInt())
/** Divides this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun div(other: UShort): UInt = this.toUInt().div(other.toUInt())
/** Divides this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun div(other: UInt): UInt = this.toUInt().div(other)
/** Divides this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun div(other: ULong): ULong = this.toULong().div(other)
/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): UInt = this.toUInt().rem(other.toUInt())
/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): UInt = this.toUInt().rem(other)
/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong = this.toULong().rem(other)
/** Calculates the remainder of dividing this value by the other value. */

@kotlin.internal.InlineOnly
public inline operator fun rangeTo(other: UShort): UIntRange = UIntRange(this.toUInt(), other.toUInt())
/** Creates a range from this value to the specified \[other\] value. */

@kotlin.internal.InlineOnly
public inline infix fun and(other: UShort): UShort = UShort(this.data and other.data)
/** Performs a bitwise AND operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun or(other: UShort): UShort = UShort(this.data or other.data)
/** Performs a bitwise OR operation between the two values. */

@kotlin.internal.InlineOnly
public inline infix fun xor(other: UShort): UShort = UShort(this.data xor other.data)
/** Performs a bitwise XOR operation between the two values. */

@kotlin.internal.InlineOnly
public inline fun inv(): UShort = UShort(data.inv())
/** Inverts the bits in this value. */

@kotlin.internal.InlineOnly
public inline fun toByte(): Byte = data.toByte()}
/** Converts this \[UShort\] value to \[Byte\] value. */

@kotlin.internal.InlineOnly
public inline fun toShort(): Short = data.toShort()}
/** Converts this \[UShort\] value to \[Short\] value. */

@kotlin.internal.InlineOnly
public inline fun toInt(): Int = data.toInt()}
/** Converts this \[UShort\] value to \[Int\] value. */

@kotlin.internal.InlineOnly
public inline fun toLong(): Long = data.toLong()}
/** Converts this \[UShort\] value to \[Long\] value. */

Open Source Used In IOx-FogDirector-AC18 1.17.0 2126
public inline fun toLong(): Long = data.toLong() and 0xFFFF

/**
 * Converts this [UShort] value to [UByte].
 * If this value is less than or equals to [UByte.MAX_VALUE], the resulting `UByte` value represents
 * the same numerical value as this `UShort`. * The resulting `UByte` value is represented by the least significant 8 bits of this `UShort`. * * The least significant 16 bits of the resulting `UInt` value are the same as the bits of this `UShort`. * whereas the most significant 16 bits are filled with zeros. * Converts this [UShort] value to [UInt]. * The resulting `UInt` value represents the same numerical value as this `UShort`. * * The least significant 16 bits of the resulting `ULong` value are the same as the bits of this `UShort`. * whereas the most significant 48 bits are filled with zeros. * Converts this [UShort] value to [ULong]. * The resulting `ULong` value represents the same numerical value as this `UShort`. * * The least significant 16 bits of the resulting `Float` value are the same as the bits of this `UShort`. * The resulting `Float` value represents the same numerical value as this `UShort`. * * The least significant 16 bits of the resulting `Double` value are the same as the bits of this `UShort`. * The resulting `Double` value represents the same numerical value as this `UShort`. * * If this value is positive, the resulting `UShort` value represents the same numerical value as this `Byte`. * * The least significant 8 bits of the resulting `UShort` value are the same as the bits of this `Byte`. * whereas the most significant 8 bits are filled with the sign bit of this value. * Converts this [Byte] value to [UShort]. * * If this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents the same numerical value as this `Int`. * * The resulting `UShort` value is represented by the least significant 16 bits of this `Int`. * * Converts this [Int] value to [UShort]. * If this value is positive and less than or equals to [UShort.MAX_VALUE], the resulting `UShort` value represents the same numerical value as this `Long`. * * The resulting `UShort` value is represented by the least significant 16 bits of this `Long`. * * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 */

package kotlin.ranges

/**
 * A range of values of type `Char`.
 */
public class CharRange(start: Char, endInclusive: Char) : CharProgression(start, endInclusive, 1), ClosedRange<Char> {
    override val start: Char get() = first
    override val endInclusive: Char get() = last
    override fun contains(value: Char): Boolean = first <= value && value <= last
    override fun isEmpty(): Boolean = first > last
    override fun equals(other: Any?): Boolean = other is CharRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last)
    override fun toString(): String = if (isEmpty()) "Empty" else last.toString()
    companion object {
        /** An empty range of values of type Char. */
        public val EMPTY: CharRange = CharRange(1.toChar(), 0.toChar())
    }
}

/**
 * A range of values of type `Int`.
 */
public class IntRange(start: Int, endInclusive: Int) : IntProgression(start,
endInclusive, 1), ClosedRange<Int> { println override val start: Int = first println override val endInclusive: Int = last println override fun contains(value: Int): Boolean = first <= value && value <= last println override fun isEmpty(): Boolean = first > last println override fun equals(other: Any?): Boolean = other is IntRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last) println override fun hashCode(): Int = if (isEmpty()) -1 else (31 * first + last) println toString(): String = "$first..$last companion object {
  /** An empty range of values of type Int. */
  public val EMPTY: IntRange = IntRange(1, 0)
}

/** A range of values of type `Long`. */
public class LongRange(start: Long, endInclusive: Long) : LongProgression(start, endInclusive, 1), ClosedRange<Long> { println override val start: Long get() = first println override val endInclusive: Long get() = last println override fun contains(value: Long): Boolean = first <= value && value <= last println override fun isEmpty(): Boolean = first > last println other is LongRange && (isEmpty() && other.isEmpty() || first == other.first && last == other.last) println override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (first xor (first ushr 32)) + (last xor (last ushr 32))).toInt() println toString(): String = "$first..$last companion object {
  /** An empty range of values of type Long. */
  public val EMPTY: LongRange = LongRange(1, 0)
}

// /*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")
@file:UseExperimental(kotlin.experimental.ExperimentalTypeInference::class)

package kotlin.collections

import kotlin.contracts.*

internal object EmptyIterator : ListIterator<Nothing> {
    override fun hasNext(): Boolean = false
    override fun hasPrevious(): Boolean = false
    override fun nextIndex(): Int = 0
    override fun previousIndex(): Int = -1
    override fun next(): Nothing = throw NoSuchElementException()
    override fun previous(): Nothing = throw NoSuchElementException()
}

internal object EmptyList : List<Nothing>, Serializable, RandomAccess {
    private const val serialVersionUID: Long = -7390468764508069838L

    override fun equals(other: Any?): Boolean = other is List<*> && other.isEmpty()
    override fun hashCode(): Int = 1
    override fun toString(): String = "[]"

    override val size: Int get() = 0
    override fun isEmpty(): Boolean = true
    override fun contains(element: Nothing): Boolean = false
    override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
    override fun get(index: Int): Nothing = throw IndexOutOfBoundsException("Empty list doesn't contain element at index $index.")
    override fun indexOf(element: Nothing): Int = -1
    override fun lastIndexOf(element: Nothing): Int = -1

    override fun iterator(): Iterator<Nothing> = EmptyIterator
    override fun listIterator(): ListIterator<Nothing> = EmptyIterator
    override fun listIterator(index: Int): ListIterator<Nothing> {
        if (index != 0) throw IndexOutOfBoundsException("Index: $index")
        return EmptyIterator
    }

    override fun subList(fromIndex: Int, toIndex: Int): List<Nothing> {
        if (fromIndex == 0 && toIndex == 0) return this
        throw IndexOutOfBoundsException("fromIndex: $fromIndex, toIndex: $toIndex")
    }

    private fun readResolve(): Any = EmptyList
}

/** Returns an empty read-only list.  The returned list is serializable (JVM). */
public fun <T> emptyList(): List<T> = EmptyList

/** Returns a new read-only list of given elements.  The returned list is serializable (JVM). */
public fun <T> listOf(vararg elements: T): List<T> = if (elements.size > 0) elements.asList() else emptyList()
new [MutableList].
 * @sample samples.collections.Collections.Lists.emptyList
 /*@SinceKotlin("1.1")*/@kotlin.internal.InlineOnly
mutable inline fun <T> mutableListOf(): MutableList<T> =
    ArrayList()

/**
 * Returns an empty new [MutableList] with the given elements.
 * @sample samples.collections.Collections.Lists.emptyList
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> mutableListOf(vararg elements: T):
    MutableList<T> =
    if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs =
true))

/**
 * Returns a empty new list either of single given element, if it is not null, or empty list if the element is null. The
returned list is serializable (JVM).
 * @sample samples.collections.Collections.Lists.emptyList
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> arrayListOf(): ArrayList<T> =
    ArrayList()

/**
 * Returns a new read-only list with the given elements.
 * @sample samples.collections.Collections.Lists.emptyList
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> MutableList(size: Int, init: (index: Int) -> T): MutableList<T> {
    val list = ArrayList<T>(size)
    repeat(size) { index -> list.add(init(index)) }
    return list
}

/**
 * Builds a new read-only 
[Array][List] by populating a [MutableList] using the given 
[builderAction] and returning a read-only list with the same elements.
 * @sample samples.collections.Collections.Lists.emptyList
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <E> buildList(@BuilderInference builderAction: MutableList<E>.() -> Unit): List<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE)
    }
    return
    ArrayList<E>().apply(builderAction)
}

/**
 * Returns an 
[Int][Range] of the valid indices for this 
collection.
 * @sample samples.collections.Collections.Collections.indicesOfCollection
 */
public val Collection<*>.indices: IntRange
    get() = 0..size - 1

/**
 * Returns the index of the last item in the list or -1 if
the list is empty.
 * @sample samples.collections.Collections.Lists.lastIndexOfList
 */
public val <T> List<T>.lastIndex: Int
    get() = this.size - 1

/**
 * Returns `true` if the collection is not empty.
 * @sample samples.collections.Collections.Collections.emptyList
 */
public val Collection<*>.isEmpty: Boolean
    get() = false

/*@SinceKotlin("1.1")*/@kotlin.internal.InlineOnly
mutable inline fun <T> arrayListOf(vararg elements: T):
    ArrayList<T> =
    if (elements.size == 0) ArrayList() else ArrayList(ArrayAsCollection(elements, isVarargs =
true))
samples.collections.Collections.Collections.isNotEmpty() *\n
/*
 * Returns this Collection if it's not `null` and the empty list otherwise.
 * @sample samples.collections.Collections.Collections.emptyList
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>?.isEmpty(): Boolean {
    contract {
        returns(false) implies (this@isEmpty == null)
    }

    return this == null ||
            this.isEmpty()
}

/**
 * Returns this Collection if it's not `null` or the result of calling 
 * [defaultValue] function if the collection is empty.
 * @sample samples.collections.Collections.Collections.emptyList
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : Collection<*>, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Searches this Collection or its range for the provided [element] using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the Comparable natural ordering of its elements.
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element, if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted,
 * so that the list (or the specified subrange of list) still remains sorted.
 * @sample samples.collections.Collections.Collections.emptyList
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> Collection<T>.indexOf(element: T?, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = compareValues(midVal, element)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}

/**
 * Searches this Collection or its range for the provided [element] using the binary search algorithm.
 * The list is expected to be sorted into ascending order according to the specified [comparator],
 * otherwise the result is undefined.
 * If the list contains multiple elements equal to the specified [element], there is no guarantee which one will be found.
 * `null` value is considered to be less than any non-null value.
 * @return the index of the element, if it is contained in the list within the specified range;
 * otherwise, the inverted insertion point `(insertion point - 1)`.
 * The insertion point is defined as the index at which the element should be inserted,
 * so that the list (or the specified subrange of list) still remains sorted.
 * @sample samples.collections.Collections.Collections.emptyList
 */
@Suppress("EXTENSION_SHADOWED_BY_MEMBER") // false warning, extension takes precedence in some cases
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.indexOf(element: T, comparator: Comparator<T>, fromIndex: Int = 0, toIndex: Int = size): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparator.compare(midVal, element)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}
Searches this list or its range for an element having the key returned by the specified [selector] function equal to the provided [key] value using the binary search algorithm. The list is expected to be sorted into ascending order according to the Comparable natural ordering of keys of its elements. otherwise the result is undefined. If the list contains multiple elements with the specified [key], there is no guarantee which one will be found. A null value is considered to be less than any non-null value. If the list is found, return the index of the element with the specified [key], if it is contained in the list within the specified range, otherwise, the inverted insertion point (-insertion point - 1). The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.

```kotlin
public inline fun <T, K : Comparable<K>>
List<T>.binarySearchByKey(key: K?, fromIndex: Int = 0, toIndex: Int = size, selector: (T) -> K?): Int =
    binarySearch(fromIndex, toIndex) { compareValues(selector(it), key) }
```

`null` value is considered to be less than any non-null value.

@return the index of the element with the specified [key], if it is contained in the list within the specified range, otherwise, the inverted insertion point (-insertion point - 1). The insertion point is defined as the index at which the element should be inserted, so that the list (or the specified subrange of list) still remains sorted.

On the elements coming before the target element, the function must return negative values; on the elements coming after the target element, the function must return positive values. If the list contains multiple elements for which [comparison] returns zero, there is no guarantee which one will be found.

```kotlin
public fun <T, K> List<T>.binarySearchBy(key: K, comparator: Comparator<K>, fromIndex: Int = 0, toIndex: Int = size, selector: (T) -> K): Int =
    binarySearch(fromIndex, toIndex) {
        comparator.compare(selector(it), key) }
```

Searches this list or its range for an element for which the given [comparison] function returns zero using the binary search algorithm. The list is expected to be sorted so that the signs of the [comparison] function's return values ascend on the list elements, i.e. negative values come before zero and zeroes come before positive values. Otherwise, the result is undefined. If the list contains multiple elements for which [comparison] returns zero, there is no guarantee which one will be found.

```kotlin
public fun <T>
List<T>.binarySearch(fromIndex: Int = 0, toIndex: Int = size, comparison: (T) -> Int): Int {
    rangeCheck(size, fromIndex, toIndex)

    var low = fromIndex
    var high = toIndex - 1

    while (low <= high) {
        val mid = (low + high).ushr(1) // safe from overflows
        val midVal = get(mid)
        val cmp = comparison(midVal)

        if (cmp < 0)
            low = mid + 1
        else if (cmp > 0)
            high = mid - 1
        else
            return mid // key found
    }
    return -(low + 1)  // key not found
}
```

Checks that `from` and `to` are in the range of [0..size] and throws an appropriate exception, if they aren't.

```kotlin
private fun rangeCheck(size: Int, fromIndex: Int, toIndex: Int) {
    when {
        fromIndex > toIndex -> throw IllegalArgumentException("fromIndex ($fromIndex) is greater than toIndex ($toIndex).")
        fromIndex < 0 -> throw IndexOutOfBoundsException("fromIndex ($fromIndex) is less than zero.")
        toIndex > size -> throw IndexOutOfBoundsException("toIndex ($toIndex) is greater than size ($size).")
    }
}
```

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

`null` private object EmptyMap : Map<*, Nothing> { override fun checkIndexOverflow() { throw ArithmeticException("Index overflow has happened.") } }, `null` private object EmptyList : List<*> { override fun checkCountOverflow() { throw ArithmeticException("Count overflow has happened.") } }, Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

Open Source Used In IOx-FogDirector-AC18 1.1.7.0 2131
Any?): Boolean = false
    override fun containsValue(value: Nothing?): Boolean = false
    override fun get(key: Any?): Nothing? = null
    override fun values(): Collection<Nothing> = EmptyList
    private fun readResolve(): Any = EmptyMap
}

/**
 * Returns an empty read-only map of specified type.
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> emptyMap(): Map<K, V> = @Suppress("UNCHECKED_CAST") (emptyMap() as Map<K, V>)

/**
 * Returns a new read-only map with the specified contents, given as a list of pairs
 * where the first value is the key and the second is the value.
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> mapOf(vararg pairs: Pair<K, V>): Map<K, V> =
    if (pairs.size > 0) mapOf(LinkedHashMap(mapCapacity(pairs.size))).apply { putAll(pairs) } else
    emptyMap() as Map<K, V>

/**
 * Returns a new read-only map.
 * @sample samples.collections.Maps.Instantiation.emptyReadOnlyMap
 */
public fun <K, V> mapOf(): Map<K, V> = emptyMap()

/**
 * Returns a new mutable map with the specified contents, given as a list of pairs
 * where the first value is the key and the second is the value.
 * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
 * @sample samples.collections.Maps.Instantiation.emptyMutableMap
 */
public fun <K, V> mutableMapOf(vararg pairs: Pair<K, V>): MutableMap<K, V> =
    LinkedHashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns a new HashMap with the specified contents, given as a list of pairs
 * where the first component is the key and the second is the value.
 * @sample samples.collections.Maps.Instantiation.hashMapFromPairs
 */
public fun <K, V> hashMapOf(vararg pairs: Pair<K, V>): HashMap<K, V> =
    HashMap<K, V>(mapCapacity(pairs.size)).apply { putAll(pairs) }

/**
 * Returns a new LinkedHashMap with the specified contents, given as a list of pairs
 * where the first component is the key and the second is the value.
 * If multiple pairs have the same key, the resulting map will contain the value from the last of those pairs.
 * @sample samples.collections.Maps.Instantiation.linkedMapFromPairs
 */
public fun <K, V> linkedMapOf(vararg pairs: Pair<K, V>): LinkedHashMap<K, V> = pairs.toMap(LinkedHashMap(mapCapacity(pairs.size)))

/**
 * Builds a new read-only map by populating a mutable map using the given builder action
 * and returning a read-only map with the same key-value pairs.
 * The map passed as a receiver to the builder action is valid only inside that function.
 * Using it outside of the function produces an unspecified behavior.
 * Entries of the map are iterated in the order they were specified.
 * @sample samples.collections.Maps.Instantiation.buildMapSample
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <K, V> buildMap(@BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) } return LinkedHashMap<K, V>(builderAction)
}

/**
 * Builds a new read-only map by populating a mutable map using the
given [builderAction]\n* and returning a read-only map with the same key-value pairs.\n* The map passed as a receiver to the [builderAction] is valid only inside that function.\n* Using it outside of the function produces an unspecified behavior.\n* [capacity] is used to hint the expected number of pairs added in the [builderAction].\n* Entries of the map are iterated in the order they were added by the [builderAction].\n* @throws IllegalArgument Exception if the given [capacity] is negative.\n* @sample samples.collections.Builders.Maps.buildMapSample
*/
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <K, V> buildMap(capacity: Int, @BuilderInference builderAction: MutableMap<K, V>.() -> Unit): Map<K, V> {
    contract { callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) }\n    checkBuilderCapacity(capacity)\n    return LinkedHashMap<K, V>(mapCapacity(expectedSize: Int)).apply(builderAction)\n* Checks a collection builder function capacity argument.\n
*\n@SinceKotlin("1.3")\n@ExperimentalStdlibApi\n@PublishedApi\ninternal expect fun checkBuilderCapacity(capacity: Int)\n
/** Returns `true` if this map is not empty.\n*\n* @throws IllegalArgumentException if the given [capacity] is negative.\n*\n* @sample samples.collections.Maps.buildMapSample\n*/
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@PublishedApi
internal expect fun sampleMapIsNullOrEmpty\n
/** Returns `true` if this map is either null or empty.\n*\n* @sample samples.collections.Maps.buildMapSample\n*/
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>?.isEmpty(): Boolean {
    contract {
        returns(false) implies (this@isEmpty != null)\n    }\n    return this == null || !isEmpty()\n}

/** Returns `true` if the map contains the specified [key].\n*\n* Allows to avoid type-safety restriction of `containsKey` that requires to pass a key of type `K`.\n*\n* @sample samples.collections.Maps.buildMapSample\n*/
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>?.contains(key: K): Boolean =
    @Suppress("UNCHECKED_CAST") (this as Map<K, Any>).containsKey(key)\n
/** Returns `true` if the map contains the specified [value].\n*\n* Allows to avoid type-safety restriction of `containsValue` that requires to pass a value of type `V`.\n*\n* @sample samples.collections.Maps.buildMapSample\n*/
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<K, V>?.containsValue(value: V): Boolean = this.containsValue(value)\n
/** Removes the specified key and its corresponding value from this map.\n*\n* @return the previous value associated with the key, or `null` if the key was not present in the map.\n*\n* @sample samples.collections.Maps.buildMapSample\n*/
@kotlin.internal.InlineOnly
public inline fun <K, V> MutableMap<K, V>.remove(key: K): V? {
    return @Suppress("UNCHECKED_CAST") (this as MutableMap<K, Any>).remove(key)\n}

// Open Source Used In IOx-FogDirector-AC18 1.17.0 2133
destructuring declarations when working with maps, for example:

```kotlin
for ((key, value) in map) {
    // do something with the key and the value
}
```

This method allows to use destructuring declarations when working with maps, for example:

```kotlin
for ((key, value) in map) {
    // do something with the key and the value
}
```

The returned map preserves the entry iteration order of the original map.
`@sample samples.collections.Maps.Transformations.mapValues\n */\n// transparency.puzzle<br>\npublic inline fun <K, V, R> Map<out K, V>.mapValues(transform: (Map.Entry<K, V>) -> R): Map<K, R> {\n    return mapValuesTo(LinkedHashMap<K, R>(mapCapacity(size)), transform) \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new Map with entries having the keys obtained by applying the [transform] function to each entry in this\n * [Map] and the values of this map.\n */\npublic inline fun <K, V, R> Map<out K, V>.mapKeys(transform: (Map.Entry<K, V>) -> R): Map<K, R> {\n    return mapKeysTo(LinkedHashMap<K, R>(mapCapacity(size)), transform) \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new Map containing all key-value pairs with values matching the given [predicate].\n */\npublic inline fun <K, V> Map<out K, V>.filter(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> {\n    return filterTo(LinkedHashMap<K, V>(), predicate) \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new map containing all key-value pairs with values matching the given [predicate].\n */\npublic inline fun <K, V> Map<out K, V>.filterNot(predicate: (Map.Entry<K, V>) -> Boolean): Map<K, V> {\n    return filterNotTo(LinkedHashMap<K, V>(), predicate) \n} // optimizeReadOnlyMap()\n
/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given collection of pairs.\n */\npublic fun <K, V> Collection<Pair<K, V>>.toMap(destination: Map<K, V>): Map<K, V> {\n    return toMap(LinkedHashMap<K, V>(mapCapacity(size))).optimizeReadOnlyMap() \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new map containing all key-value pairs matching the given [predicate].\n */\npublic inline fun <K, V> Map<out K, V>.filterTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {\n    return filterTo(LinkedHashMap<K, V>(), predicate) \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new map containing all key-value pairs with values matching the given [predicate].\n */\npublic inline fun <K, V> Map<out K, V>.filterNotTo(destination: M, predicate: (Map.Entry<K, V>) -> Boolean): M {\n    return filterNotTo(LinkedHashMap<K, V>(), predicate) \n} // optimizeReadOnlyMap()\n
/**\n * Returns a new map containing all key-value pairs from the given collection of pairs.\n */\npublic fun <K, V> Iterable<Pair<K, V>>.toMap(): Map<K, V> {\n    if (this is Collection) {\n        return when (size) {\n            0 -> emptyMap() \n            1 -> mapOf(if (this is List) this[0] else iterator().next()) \n            else -> toMap(LinkedHashMap<K, V>(mapCapacity(size))) \n        } \n    } \n    return toMap(LinkedHashMap<K, V>()).optimizeReadOnlyMap() \n} // optimizeReadOnlyMap()\n
/**\n * Populates and returns the [destination] mutable map with key-value pairs from the given collection of pairs.\n */\npublic fun <K, V> Array<out Pair<K, V>>.toMap(destination: M): M {\n    return putAll(this@toMap) \n} // optimizeReadOnlyMap()\n
/\n*/\n
from the given array of pairs.

```kotlin
    destination.apply { putAll(this@toMap) } // Returns a new map containing all key-value pairs from the given sequence of pairs.

public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>(this)).optimizeReadOnlyMap() // Populates and returns the [destination] mutable map with key-value pairs from the given sequence of pairs.

public fun <K, V> Sequence<Pair<K, V>>.toMap(): Map<K, V> = toMap(LinkedHashMap<K, V>(this)).optimizeReadOnlyMap() // Returns a new read-only map containing all key-value pairs from the original map.

The returned map preserves the entry iteration order of the original map.

@kotlin.internal.InlineOnly
    this.put(pair.first, pair.second) // Creates a new read-only map by replacing or adding an entry to this map from a given key-value [pair].

@kotlin.internal.InlineOnly
    Map<K, V> =
    LinkedHashMap(this).apply { putAll(pairs) } // Creates a new read-only map by replacing or adding entries to this map from a given array of key-value [pairs].

@kotlin.internal.InlineOnly
    LinkedHashMap(this).apply { putAll(map) } // Creates a new read-only map by replacing or adding entries to this map from another [map].
```

Open Source Used in IOx-FogDirector-AC18 1.17.0 2136
containing all entries of the original map except the entry with the given [key].

* The returned map preserves the entry iteration order of the original map.

* @ SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(key) }

* @ SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }

* @ SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }

* @ SinceKotlin("1.1")
    this.toMutableMap().apply { minusAssign(keys) }

* @ SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(key: K) {
    remove(key)
}

* @ SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Iterable<K>) {
    this.keys.removeAll(keys)
}

* @ SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Array<out K>) {
    this.keys.removeAll(keys)
}

* @ SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline operator fun <K, V> MutableMap<K, V>.minusAssign(keys: Sequence<K>) {
    this.keys.removeAll(keys)
}

* @ PublishedApi
internal fun <K, V> Map<out K, V>.optimizeReadOnlyMap() = when (size) {
    0 -> emptyMap()
    1 -> toSingletonMapOrSelf()
    else -> this
}

* @file:kotlin.jvm.JvmMultifileClass
* @file:kotlin.jvm.JvmName("SetsKt")
* @file:UseExperimental(kotlin.experimental.ExperimentalTypeInference::class)

*package kotlin.collections

*import kotlin.contracts.*

*internal object EmptySet : Set<Nothing>, Serializable {
    private const val serialVersionUID: Long = 3406603774387020532

    *override fun equals(other: Any?): Boolean = other is Set<*> && other.isEmpty()
    *override fun hashCode(): Int = 0
    *override fun toString(): String = "["]

    *override val size: Int get() = 0
    *override fun isEmpty(): Boolean = true
    *override fun contains(element: Nothing): Boolean = false
    *override fun containsAll(elements: Collection<Nothing>): Boolean = elements.isEmpty()
    *override fun iterator(): Iterator<Nothing> = EmptyIterator

    *private fun readResolve(): Any = EmptySet
}

*public fun <T> emptySet(): Set<T> = EmptySet

*public fun <T> setOf(vararg elements: T): Set<T> = elements.toSet()

*public fun <T> emptySet(): Set<T> = EmptySet

*public fun <T> setOf(): Set<T> = emptySet()
samples.collections.Collections.Sets.emptyMutableSet

@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> mutableSetOf(): MutableSet<T> = elements.toCollection(LinkedHashSet(mapCapacity(elements.size)))

/**
 * Returns a new [HashSet] with the given elements.
 * Elements of the set are iterated in the order they were specified.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> hashSetOf(): HashSet<T> =

/**
 * Returns a new [HashSet] with the given elements.
 * Elements of the set are iterated in the order they were specified.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun <T> linkedSetOf(): LinkedHashSet<T> =

/**
 * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]
 * and returning a read-only set with the same elements.
 * The set passed as a receiver to the [builderAction] is valid only inside that function.
 * Using it outside of the function produces an unspecified behavior.
 * Elements of the set are iterated in the order they were added by the [builderAction].
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <E> buildSet(@BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) } 
    return LinkedHashSet<E>().apply(builderAction)
}

/**
 * Builds a new read-only [Set] by populating a [MutableSet] using the given [builderAction]
 * and returning a read-only set with the same elements.
 * The set passed as a receiver to the [builderAction] is valid only inside that function.
 * Using it outside of the function produces an unspecified behavior.
 * Elements of the set are iterated in the order they were added by the [builderAction].
 * @throws IllegalArgumentException if the given [capacity] is negative.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun <E> buildSet(capacity: Int, @BuilderInference builderAction: MutableSet<E>.() -> Unit): Set<E> {
    contract {
        callsInPlace(builderAction, InvocationKind.EXACTLY_ONCE) 
    } 
    checkBuilderCapacity(capacity)
    return LinkedHashSet<E>(mapCapacity(capacity)).apply(builderAction)
}

/**
 * Parses the string as a signed [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(): Byte? = toByteOrNull(radix = 10)

/**
 * Parses the string as a signed [Byte] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 * @throws IllegalArgumentException when [radix] is not a valid radix for string to number
 * conversion.
 */
@SinceKotlin("1.1")
public fun String.toByteOrNull(radix: Int): Byte? {
    val int = this.toIntOrNull(radix) ?: return null
    if (int < Byte.MIN_VALUE || int > Byte.MAX_VALUE) return null
    return int.toByte()
}

/**
 * Parses the string as a [Short] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.1")
public fun String.toShortOrNull(): Short? =
toShortOrNull(radix = 10)

/**
 * Parses the string as a [Short] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 * @throws IllegalArgument Exception when [radix] is not a valid radix for string to number conversion.
 */
@SinceKotlin("1.1")
public fun String.toShortOrNull(radix: Int): Short? {
    val int = this.toIntOrNull(radix) ?: return null
    if (int < Short.MIN_VALUE || int > Short.MAX_VALUE) return null
    return int.toShort()
}

/**
 * Parses the string as an [Int] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.1")
public fun String.toIntOrNull(radix: Int): Int? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after
        start = 1
        if (firstChar == '-') {
            isNegative = true
        } else if (firstChar == '+') {
            isNegative = false
        } else
            return null
    } else {
        start = 0
        isNegative = false
        limit = -Int.MAX_VALUE
    }

    val limitForMaxRadix = (-Int.MAX_VALUE) / 36
    var result = 0L
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
                if (result < limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }
        result *= radix
        if (result < limit + digit)
            return null
        result -= digit
    }

    return if (isNegative) result else -result
}

/**
 * Parses the string as a [Long] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.1")
public fun String.toLongOrNull(radix: Int): Long? {
    checkRadix(radix)

    val length = this.length
    if (length == 0) return null
    if (firstChar < '0') {  // Possible leading sign
        if (length == 1) return null  // non-digit (possible sign) only, no digits after
        start = 1
        if (firstChar == '-') {
            isNegative = true
            limit = Long.MIN_VALUE
        } else if (firstChar == '+') {
            isNegative = false
            limit = -Long.MAX_VALUE
        } else
            return null
    } else {
        start = 0
        isNegative = false
        limit = -Long.MAX_VALUE
    }

    val limitForMaxRadix = (-Long.MAX_VALUE) / 36
    var result = 0L
    for (i in start until length) {
        val digit = digitOf(this[i], radix)
        if (digit < 0) return null
        if (result < limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / radix
                if (result < limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }
        result *= radix
        if (result < limit + digit)
            return null
        result -= digit
    }

    return if (isNegative) result else -result
}

internal fun numberFormatError(input: String): Nothing = throw NumberFormatException("Invalid number format: '$input'")

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file.
 */
Authors\n
internal fun ulongDivide(v1: ULong, v2: ULong): ULong {\n    val dividend = v1.toLong()\n    val divisor = v2.toLong()\n    if (divisor < 0) { // i.e., divisor >= 2^63\n        return if (v1 < v2)\n            ULong(0) else ULong(1)\n    }\n
    // Optimization - use signed division if both dividend and divisor < 2^63\n    if (dividend >= 0) {\n        return dividend / divisor\n    }\n
    // Otherwise, approximate the quotient, check, and correct if necessary.\n    val quotient = ((dividend ushr 1) / divisor) shl 1\n    val rem = dividend - quotient * divisor\n
    return ULong(quotient + if (ULong(rem) >= ULong(divisor)) 1 else 0)\n}

internal fun ulongRemainder(v1: ULong, v2: ULong): ULong {\n    val dividend = v1.toLong()\n    val divisor = v2.toLong()\n    if (divisor < 0) { // i.e., divisor >= 2^63\n        return if (v1 < v2)\n            v1 // dividend < divisor\n        else\n            v1 - v2 // dividend > divisor\n    }\n
    // Optimization - use signed modulus if both dividend and divisor < 2^63\n    if (dividend >= 0) {\n        return dividend % divisor\n    }\n
    // Otherwise, approximate the quotient, check, and correct if necessary.\n    val quotient = ((dividend ushr 1) / divisor) shl 1\n    val rem = dividend - quotient * divisor\n
    return ULong(rem - if (ULong(rem) >= ULong(divisor)) divisor else 0)\n}

internal fun doubleToUInt(v: Double): UInt = when {\n    v.isNaN() -> 0u\n    v <= UInt.MIN_VALUE.toDouble() -> UInt.MIN_VALUE\n    v >= UInt.MAX_VALUE.toDouble() -> UInt.MAX_VALUE\n    v <= Int.MAX_VALUE.toDouble() -> v.toInt().toUInt()\n    else -> (v - Int.MAX_VALUE).toInt().toUInt() + Int.MAX_VALUE.toUInt()\n}

internal fun doubleToULong(v: Double): ULong = when {\n    v.isNaN() -> 0u\n    v <= ULong.MIN_VALUE.toDouble() -> ULong.MIN_VALUE\n    v >= ULong.MAX_VALUE.toDouble() -> ULong.MAX_VALUE\n    v < Long.MAX_VALUE -> v.toLong().toULong()\n    else -> (v - 2048).toLong().toULong() + 2048uL\n}

internal fun uintToDouble(v: Int): Double = (v and Int.MAX_VALUE).toDouble() + (v ushr 31 shl 30).toDouble() * 2

internal fun ulongToDouble(v: Long): Double = (v ushr 11).toDouble() * 2048 + (v and 2047)

internal fun ulongToString(v: Long): String = ulongToString(v, 10)

internal fun ulongToString(v: Long, base: Int): String {\n    if (v >= 0) return v.toString(base)\n    var quotient = ((v ushr 1) / base) shl 1\n    var rem = v - quotient * base\n    if (rem >= base) {\n        rem -= base\n        quotient += 1\n    }\n    return quotient.toString(base) + rem.toString(base)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language
 * contributors. Use of this source code is governed by the Apache 2.0 license that can be found in
 * LICENSE.txt file.

 * File: $file
 * @file: kotlin.jvm.JvmMultifileClass
 * @file: kotlin.jvm.JvmName("CollectionsKt")
 * @package kotlin.collections
 * @* Given an [iterator] function constructs an [Iterable] instance that returns values through
 * the [Iterator] * provided by that function.
 * @* @sample sample.collections.Enumerables.Building
 * @* @* *
 * @* @* @* InlineOnly
 * @* @* @* public inline fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
 * @* @* @* override fun iterator(): Iterator<T> = iterator()\n
 * @* private class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
 * @* override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())

 * @* private fun <T> Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList\n
 * @* internal fun <T> Collection<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> = when (this) {
 * @* is Set -> this\n * @* else -> null\n
 * @* @* @* internal fun Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList\n
 * @* internal fun <T> Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList\n
 * @* convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> = when (this) {
 * @* is Collection -> this\n * @* else -> null\n
 * @* return Collections.CollectionsKt
 * @* package kotlin.collections
 * @* @* * Given an [iterator] function constructs an [Iterable] instance that returns values through
 * the [Iterator] * provided by that function.
 * @* * * * @sample sample.collections.Enumerables.Building
 * @* * * * InlineOnly
 * @* * * public inline fun <T> Iterable(crossinline iterator: () -> Iterator<T>): Iterable<T> = object : Iterable<T> {
 * @* * * override fun iterator(): Iterator<T> = iterator()\n
 * @* * private class IndexingIterable<out T>(private val iteratorFactory: () -> Iterator<T>) : Iterable<IndexedValue<T>> {
 * @* * override fun iterator(): Iterator<IndexedValue<T>> = IndexingIterator(iteratorFactory())

 * @* * private fun <T> Collection<T>.safeToConvertToSet() = size > 2 && this is ArrayList\n
 * @* * internal fun <T> Collection<T>.convertToSetForSetOperationWith(source: Iterable<T>): Collection<T> = when (this) {
 * @* * is Set -> this\n * @* * else -> null\n
 * @* * return Collections.CollectionsKt
 * @* */
 * }*/


else -> if (this.safeToConvertToSet()) toHashSet() else this\n
}\n
else -> toHashSet()\n }\n\n/**
Converts this collection to a set, when it's worth so and it doesn't change contains method behavior. */\ninternal fun
<T> Iterable<T>.convertToSetForSetOperation(): Collection<T> =\n when (this) {\n
is Set -> this\n
is
Collection -> if (this.safeToConvertToSet()) toHashSet() else this\n
else -> toHashSet()\n }\n\n\n/**\n *
Returns a single list of all elements from all collections in the given collection.\n * @sample
samples.collections.Iterables.Operations.flattenIterable\n */\npublic fun <T> Iterable<Iterable<T>>.flatten():
List<T> {\n val result = ArrayList<T>()\n for (element in this) {\n
result.addAll(element)\n }\n return
result\n}\n\n/**\n * Returns a pair of lists, where\n * *first* list is built from the first values of each pair from this
collection,\n * *second* list is built from the second values of each pair from this collection.\n * @sample
samples.collections.Iterables.Operations.unzipIterable\n */\npublic fun <T, R> Iterable<Pair<T, R>>.unzip():
Pair<List<T>, List<R>> {\n val expectedSize = collectionSizeOrDefault(10)\n val listT =
ArrayList<T>(expectedSize)\n val listR = ArrayList<R>(expectedSize)\n for (pair in this) {\n
listT.add(pair.first)\n
listR.add(pair.second)\n }\n return listT to listR\n}\n","/*\n * Copyright 2010-2018
JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the
Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmMultifileClass\n@file:kotlin.jvm.JvmName(\"SequencesKt\")\n\npackage
kotlin.sequences\n\n/**\n * Given an [iterator] function constructs a [Sequence] that returns values through the
[Iterator]\n * provided by that function.\n * The values are evaluated lazily, and the sequence is potentially
infinite.\n *\n * @sample samples.collections.Sequences.Building.sequenceFromIterator\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence(crossinline iterator: () -> Iterator<T>):
Sequence<T> = object : Sequence<T> {\n override fun iterator(): Iterator<T> = iterator()\n}\n\n/**\n * Creates a
sequence that returns all elements from this iterator. The sequence is constrained to be iterated only once.\n *\n *
@sample samples.collections.Sequences.Building.sequenceFromIterator\n */\npublic fun <T>
Iterator<T>.asSequence(): Sequence<T> = Sequence { this }.constrainOnce()\n\n/**\n * Creates a sequence that
returns the specified values.\n *\n * @sample samples.collections.Sequences.Building.sequenceOfValues\n
*/\npublic fun <T> sequenceOf(vararg elements: T): Sequence<T> = if (elements.isEmpty()) emptySequence() else
elements.asSequence()\n\n/**\n * Returns an empty sequence.\n */\npublic fun <T> emptySequence():
Sequence<T> = EmptySequence\n\nprivate object EmptySequence : Sequence<Nothing>,
DropTakeSequence<Nothing> {\n override fun iterator(): Iterator<Nothing> = EmptyIterator\n override fun
drop(n: Int) = EmptySequence\n override fun take(n: Int) = EmptySequence\n}\n\n/**\n * Returns this sequence if
it's not `null` and the empty sequence otherwise.\n * @sample
samples.collections.Sequences.Usage.sequenceOrEmpty\n
*/\n@SinceKotlin(\"1.3\")\n@kotlin.internal.InlineOnly\npublic inline fun <T> Sequence<T>?.orEmpty():
Sequence<T> = this ?: emptySequence()\n\n\n/**\n * Returns a sequence that iterates through the elements either of
this sequence\n * or, if this sequence turns out to be empty, of the sequence returned by [defaultValue] function.\n
*\n * @sample samples.collections.Sequences.Usage.sequenceIfEmpty\n */\n@SinceKotlin(\"1.3\")\npublic fun
<T> Sequence<T>.ifEmpty(defaultValue: () -> Sequence<T>): Sequence<T> = sequence {\n val iterator =
this@ifEmpty.iterator()\n if (iterator.hasNext()) {\n
yieldAll(iterator)\n } else {\n
yieldAll(defaultValue())\n }\n}\n\n/**\n * Returns a sequence of all elements from all sequences in this
sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n */\npublic fun <T>
Sequence<Sequence<T>>.flatten(): Sequence<T> = flatten { it.iterator() }\n\n/**\n * Returns a sequence of all
elements from all iterables in this sequence.\n *\n * The operation is _intermediate_ and _stateless_.\n
*/\n@kotlin.jvm.JvmName(\"flattenSequenceOfIterable\")\npublic fun <T> Sequence<Iterable<T>>.flatten():
Sequence<T> = flatten { it.iterator() }\n\nprivate fun <T, R> Sequence<T>.flatten(iterator: (T) -> Iterator<R>):
Sequence<R> {\n if (this is TransformingSequence<*, *>) {\n
return (this as TransformingSequence<*,
T>).flatten(iterator)\n }\n return FlatteningSequence(this, { it }, iterator)\n}\n\n/**\n * Returns a pair of lists,
where\n * *first* list is built from the first values of each pair from this sequence,\n * *second* list is built from the
second values of each pair from this sequence.\n *\n * The operation is _terminal_.\n */\npublic fun <T, R>

Open Source Used In IOx-FogDirector-AC18 1.17.0 2141


Sequence<Pair<T, R>>>.unzip(): Pair<List<T>, List<R>>
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

@Suppress("UNCHECKED_CAST")
    return result as T
}

override fun hasNext(): Boolean
    return iterator.hasNext()

    override fun next(): R {
        return transformer(iterator.next())
    }

private val transformer: (T) -> R
constructor(private val sequence: Sequence<T>, private val transformer: (Int, T) -> R) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        var index = 0
        override fun next(): R {
            return transformer(iterator.next())
        }
        override fun hasNext(): Boolean
            return iterator.hasNext()
    }
}

/**
 * A sequence which combines values from the underlying [sequence] with their indices and returns them as
 * IndexedValue<T> objects.
 * @param sendWhen If `true`, values for which the predicate returns `true` are returned.
 * Otherwise, values for which the predicate returns `false` are returned.
 * @param predicate The predicate to filter the values.
 */
internal class TransformingIndexedSequence<T, R, E>
constructor(private val sequence: Sequence<T>, private val transformer: (Int, T) -> R) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        var index = 0
        override fun next(): R {
            return transformer(index++, iterator.next())
        }
        override fun hasNext(): Boolean
            return iterator.hasNext()
    }
}

/**
 * A sequence that returns the values from the underlying [sequence] that either match or do not
 * match `null` * the specified [predicate].
 * @param sendWhen If `true`, values for which the predicate returns `true` are returned.
 * Otherwise, values for which the predicate returns `false` are returned.
 * @param predicate The predicate to filter the values.
 */
internal class FilteringSequence<T, R>(
    private val sequence: Sequence<T>,
    private val sendWhen: Boolean = true,
    private val predicate: (T) -> Boolean
) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
        var nextItem: T? = null
        override fun hasNext(): Boolean
            return iterator.hasNext()
        override fun next(): T {
            if (nextState == 0)
                throw NoSuchElementException()
            if (nextState == 1)
                return nextItem
            nextState = 1
            return returnState ?? 0
        }
        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which returns the results of applying the given [transformer] function to the values in the underlying
 * sequence along with the value itself.
 */
internal class TransformingSequence<T, R, E>
constructor(private val sequence: Sequence<T>, private val transformer: (T) -> R) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
        var nextItem: T? = null

        private fun calcNext() {
            while (iterator.hasNext()) {
                if (!ensureItemIterator())
                    throw NoSuchElementException()
                return itemIterator!!.next()
            }
        }

        override fun hasNext(): Boolean
            return ensureItemIterator() && itemIterator.hasNext()
        override fun next(): T {
            if (nextState == -1)
                calcNext()
            if (nextState == 0)
                throw NoSuchElementException()
            return itemIterator!!.next()
        }

        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which returns the values returned by that function. The sequence stops returning values as soon as one of the
 * underlying sequences stops returning values.
 * @param transformer The function to transform the values.
 */
internal class TransformSequence<T, R, E>
constructor(private val sequence: Sequence<T>, private val transformer: (T) -> E) : Sequence<E> {
    override fun iterator(): Iterator<E> = object : Iterator<E> {
        val iterator = sequence.iterator()
        var nextState = 1
        override fun hasNext(): Boolean
            return iterator.hasNext()
        override fun next(): E {
            return transformer(iterator.next())
        }
        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which merges the values from two parallel underlying sequences, passes them to the given [transform] function and
 * returns the values returned by that function. The sequence stops returning values as soon as one of the
 * underlying sequences stops returning values.
 * @param transform The function to transform the values.
 */
internal class MergingSequence<T1, T2, V>
constructor(private val sequence1: Sequence<T1>, private val sequence2: Sequence<T2>, private val transform: (T1, T2) -> V) : Sequence<V> {
    override fun iterator(): Iterator<V> = object : Iterator<V> {
        val iterator1 = sequence1.iterator()
        val iterator2 = sequence2.iterator()
        override fun hasNext(): Boolean
            return iterator1.hasNext() && iterator2.hasNext()
        override fun next(): V {
            if (nextState == -1)
                calcNext()
            if (nextState == 0)
                throw NoSuchElementException()
            return transform(iterator1.next(), iterator2.next())
        }
        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which flattens the values from two parallel underlying sequences, passes them to the given [transform] function and
 * returns the values returned by that function. The sequence stops returning values as soon as one of the
 * underlying sequences stops returning values.
 * @param transformer The function to transform the values.
 */
internal class FlatteningSequence<T1, T2, E>
constructor(private val sequence: Sequence<T1>, private val transformer: (T1) -> Iterator<E>) : Sequence<E> {
    override fun iterator(): Iterator<E> = object : Iterator<E> {
        val iterator1 = sequence.iterator()
        val iterator2 = transformer(iterator1.next())
        override fun hasNext(): Boolean
            return iterator1.hasNext() && iterator2.hasNext()
        override fun next(): E {
            return transformer(iterator1.next(), iterator2.next())
        }
        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which returns the values from the underlying [sequence] that either match or do not
 * match `null` * the specified [predicate].
 * @param sendWhen If `true`, values for which the predicate returns `true` are returned.
 * Otherwise, values for which the predicate returns `false` are returned.
 * @param predicate The predicate to filter the values.
 */
internal class FilteringSequence<T, R>(
    private val sequence: Sequence<T>,
    private val sendWhen: Boolean = true,
    private val predicate: (T) -> Boolean
) : Sequence<R> {
    override fun iterator(): Iterator<R> = object : Iterator<R> {
        val iterator = sequence.iterator()
        var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue
        var nextItem: T? = null

        private fun calcNext() {
            while (iterator.hasNext()) {
                if (!ensureItemIterator())
                    throw NoSuchElementException()
                return itemIterator!!.next()
            }
        }

        override fun hasNext(): Boolean
            return ensureItemIterator() && itemIterator.hasNext()
        override fun next(): T {
            if (nextState == -1)
                calcNext()
            if (nextState == 0)
                throw NoSuchElementException()
            return itemIterator!!.next()
        }

        override fun nextState(): Int
            return nextState
    }
}

/**
 * A sequence which merges the values from two parallel underlying sequences, passes them to the given [transform] function and
 * returns the values returned by that function. The sequence stops returning values as soon as one of the
 * underlying sequences stops returning values.
 * @param transform The function to transform the values.
 */
internal class MergingSequence<T1, T2, V>
constructor(private val sequence1: Sequence<T1>, private val sequence2: Sequence<T2>, private val transform: (T1, T2) -> V) : Sequence<V> {
    override fun iterator(): Iterator<V> = object : Iterator<V> {
        val iterator1 = sequence1.iterator()
        val iterator2 = transform(iterator1.next())
        override fun hasNext(): Boolean
            return iterator1.hasNext() && iterator2.hasNext()
        override fun next(): V {
            if (nextState == -1)
                calcNext()
            if (nextState == 0)
                throw NoSuchElementException()
            return transform(iterator1.next(), iterator2.next())
        }
        override fun nextState(): Int
            return nextState
    }
}
while (itemIterator == null) {\n
if (!iterator.hasNext()) {\n
return false\n
} else {\n
val element = iterator.next()\n
val nextItemIterator = iterator(transformer(element))\n
if (nextItemIterator.hasNext()) {\n
itemIterator = nextItemIterator\n
return true\n
}\n
}\n
}\n
return true\n
}\n }\n}\n\n/**\n * A sequence that supports drop(n) and
take(n) operations\n */\ninternal interface DropTakeSequence<T> : Sequence<T> {\n fun drop(n: Int):
Sequence<T>\n fun take(n: Int): Sequence<T>\n}\n\n/**\n * A sequence that skips [startIndex] values from the
underlying [sequence]\n * and stops returning values right before [endIndex], i.e. stops at `endIndex - 1`\n
*/\ninternal class SubSequence<T>(\n private val sequence: Sequence<T>,\n private val startIndex: Int,\n
private val endIndex: Int\n) : Sequence<T>, DropTakeSequence<T> {\n\n init {\n
require(startIndex >= 0) {
\"startIndex should be non-negative, but is $startIndex\" }\n
require(endIndex >= 0) { \"endIndex should be
non-negative, but is $endIndex\" }\n
require(endIndex >= startIndex) { \"endIndex should be not less than
startIndex, but was $endIndex < $startIndex\" }\n }\n\n private val count: Int get() = endIndex - startIndex\n\n
override fun drop(n: Int): Sequence<T> = if (n >= count) emptySequence() else SubSequence(sequence, startIndex
+ n, endIndex)\n override fun take(n: Int): Sequence<T> = if (n >= count) this else SubSequence(sequence,
startIndex, startIndex + n)\n\n override fun iterator() = object : Iterator<T> {\n\n
val iterator =
sequence.iterator()\n
var position = 0\n\n
// Shouldn't be called from constructor to avoid premature
iteration\n
private fun drop() {\n
while (position < startIndex && iterator.hasNext()) {\n
iterator.next()\n
position++\n
}\n
}\n\n
override fun hasNext(): Boolean {\n
drop()\n
return (position < endIndex) && iterator.hasNext()\n
}\n\n
override fun next(): T {\n
drop()\n
if (position >= endIndex)\n
throw NoSuchElementException()\n
position++\n
return
iterator.next()\n
}\n }\n}\n\n/**\n * A sequence that returns at most [count] values from the underlying
[sequence], and stops returning values\n * as soon as that count is reached.\n */\ninternal class TakeSequence<T>(\n
private val sequence: Sequence<T>,\n private val count: Int\n) : Sequence<T>, DropTakeSequence<T> {\n\n
init {\n
require(count >= 0) { \"count must be non-negative, but was $count.\" }\n }\n\n override fun drop(n:
Int): Sequence<T> = if (n >= count) emptySequence() else SubSequence(sequence, n, count)\n override fun
take(n: Int): Sequence<T> = if (n >= count) this else TakeSequence(sequence, n)\n\n override fun iterator():
Iterator<T> = object : Iterator<T> {\n
var left = count\n
val iterator = sequence.iterator()\n\n
override
fun next(): T {\n
if (left == 0)\n
throw NoSuchElementException()\n
left--\n
return
iterator.next()\n
}\n\n
override fun hasNext(): Boolean {\n
return left > 0 && iterator.hasNext()\n
}\n }\n}\n\n/**\n * A sequence that returns values from the underlying [sequence] while the [predicate] function
returns\n * `true`, and stops returning values once the function returns `false` for the next element.\n */\ninternal
class TakeWhileSequence<T>\nconstructor(\n private val sequence: Sequence<T>,\n private val predicate: (T) > Boolean\n) : Sequence<T> {\n override fun iterator(): Iterator<T> = object : Iterator<T> {\n
val iterator =
sequence.iterator()\n
var nextState: Int = -1 // -1 for unknown, 0 for done, 1 for continue\n
var nextItem: T?
= null\n\n
private fun calcNext() {\n
if (iterator.hasNext()) {\n
val item = iterator.next()\n
if (predicate(item)) {\n
nextState = 1\n
nextItem = item\n
return\n
}\n
}\n
nextState = 0\n
}\n\n
override fun next(): T {\n
if (nextState == -1)\n
calcNext() // will change nextState\n
if (nextState == 0)\n
throw NoSuchElementException()\n
@Suppress(\"UNCHECKED_CAST\")\n
val result = nextItem as T\n\n
// Clean next to avoid keeping
reference on yielded instance\n
nextItem = null\n
nextState = -1\n
return result\n
}\n\n
override fun hasNext(): Boolean {\n
if (nextState == -1)\n
calcNext() // will change nextState\n
return nextState == 1\n
}\n }\n}\n\n/**\n * A sequence that skips the specified number of values from the
underlying [sequence] and returns\n * all values after that.\n */\ninternal class DropSequence<T>(\n private val
sequence: Sequence<T>,\n private val count: Int\n) : Sequence<T>, DropTakeSequence<T> {\n init {\n
require(count >= 0) { \"count must be non-negative, but was $count.\" }\n }\n\n override fun drop(n: Int):
Sequence<T> = (count + n).let { n1 -> if (n1 < 0) DropSequence(this, n) else DropSequence(sequence, n1) }\n
override fun take(n: Int): Sequence<T> = (count + n).let { n1 -> if (n1 < 0) TakeSequence(this, n) else
SubSequence(sequence, count, n1) }\n\n override fun iterator(): Iterator<T> = object : Iterator<T> {\n
val

Open Source Used In IOx-FogDirector-AC18 1.17.0 2143


iterator = sequence.iterator()
var left = count

// Shouldn't be called from constructor to avoid premature iteration
private fun drop() {
    if (left > 0 && iterator.hasNext())
        return
    iterator.next()
    left--
}

override fun override fun next(): T {
    drop()
    return iterator.next()
}

override fun hasNext(): Boolean {
    drop()
    return iterator.hasNext()
}

/**
 * A sequence that skips the values from the underlying [sequence] while the given
 * [predicate] returns `true` and returns all values after that.
 */
internal class DropWhileSequence<T>(
    private val sequence: Sequence<T>,
    private val predicate: (T) -> Boolean
) : Sequence<T> {

    override fun iterator(): Iterator<T> = object : Iterator<T> {
        val iterator = sequence.iterator()
        var dropState: Int = -1 // -1 for not dropping, 1 for nextItem, 0 for normal iteration
        var nextItem: T? = null

        private fun drop() {
            while (iterator.hasNext()) {
                val item = iterator.next()
                if (!predicate(item)) {
                    nextItem = item
                    dropState = 1
                    return
                }
            }
            dropState = 0
        }

        override fun next(): T {
            if (dropState == -1)
                drop()

            if (dropState == 1) {
                @Suppress("UNCHECKED_CAST")
                val result = nextItem as T
                nextItem = null
                dropState = 0
                return result
            }
            return iterator.next()
        }

        override fun hasNext(): Boolean {
            if (dropState == -1)
                drop()
            return dropState == 1 || iterator.hasNext()
        }
    }
}

/**
 * A sequence that returns values from the underlying [sequence] after the first value
 * for which the given [predicate] returns `false`. The return value is dropped on the next
 * iteration.
 */
internal class DistinctSequence<T, K>(
    private val source: Sequence<T>,
    private val keySelector: (T) -> K
) : Sequence<T> {
    override fun iterator(): Iterator<T> = DistinctIterator(source.iterator(), keySelector)
}

private class DistinctIterator<T, K>(
    private val source: Iterator<T>,
    private val keySelector: (T) -> K)

    : AbstractIterator<T>() {

        private val observed = HashSet<K>()

        override fun computeNext() {
            while (source.hasNext()) {
                val next = source.next()
                val key = keySelector(next)

                if (observed.add(key)) {
                    setNext(next)
                    return
                }
            }

            done()
        }
    }
}

/**
 * A sequence that returns values from the underlying [sequence] after the first value
 * for which the given [predicate] returns `false`. The return value is dropped on the next
 * iteration.
 */
private class GeneratorSequence<T : Any>(
    private val getInitialValue: () -> T?,
    private val getNextValue: (T) -> T?)

    : Sequence<T> {

    override fun iterator(): Iterator<T> = object : Iterator<T> {

        var nextItem: T? = null
        var nextState: Int = -2 // -2 for initial unknown, -1 for next unknown, 0 for done, 1 for continue

        private fun calcNext() {
            nextItem = if (nextState == -2) getInitialValue() else getNextValue(nextItem!!)
            nextState = if (nextItem == null) 0 else 1
        }

        override fun next(): T {
            if (nextState < 0)
                calcNext()

            if (nextState == 0)
                throw NoSuchElementException()
            val result = nextItem as T
            nextState = -1
            return result
        }

        override fun hasNext(): Boolean {
            if (nextState < 0)
                calcNext()
            return nextState == 1
        }
    }
}

/**
 * Returns a wrapper sequence that provides
 * values of this sequence, but ensures it can be iterated only one time.
 * The operation is _intermediate_ and _stateless_.
 * The returned sequence is constrained to be iterated only once.
 */
public fun <T> Sequence<T>.constrainOnce(): Sequence<T> {
    return if (this is ConstrainedOnceSequence<T>)
        this
    else ConstrainedOnceSequence<T>(this)
}

/**
 * Returns a sequence which
 * invokes the function to calculate the next value on each iteration until the function returns `null`.
 * The returned sequence is constrained to be iterated only once.
 */
@kotlin.internal.LowPriorityInOverloadResolution
public fun <T : Any> generateSequence(nextFunction: () -> T?): Sequence<T> {
    return GeneratorSequence(nextFunction, { nextFunction() }).constrainOnce()
}

// as? does not work in js
//return this as?

nextFunction: (T) -> T?: Sequence<T> =\n    if (seed == null)\n        EmptySequence\n    else\n    GeneratorSequence({ seed }, nextFunction)\n
/**
 * Returns a sequence defined by the function [seedFunction],
 * which is invoked to produce the starting value,\n * and the [nextFunction], which is invoked to calculate the next
 * value based on the previous one on each iteration.\n * The sequence produces values until it encounters first
 * `null` value.\n * If [seedFunction] returns `null`, an empty sequence is produced.\n * The sequence can be iterated multiple times,\n */

/**
 * Returns a sequence defined by the function [seedFunction],
 * which is invoked to produce the starting value,\n * and the [nextFunction], which is invoked to calculate the next
 * value based on the previous one on each iteration.\n * The sequence produces values until it encounters first
 * `null` value.\n * If [seedFunction] returns `null`, an empty sequence is produced.\n * The sequence can be iterated multiple times,\n */
package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED
// by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.js.*
import primitiveArrayConcat
import withType
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun <T> Array<out T>.elementAt(index: Int): T {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun ByteArray.elementAt(index: Int): Byte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun ShortArray.elementAt(index: Int): Short {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun IntArray.elementAt(index: Int): Int {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun LongArray.elementAt(index: Int): Long {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun FloatArray.elementAt(index: Int): Float {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun DoubleArray.elementAt(index: Int): Double {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun BooleanArray.elementAt(index: Int): Boolean {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given 
 * or throws an
 * [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample
 * samples.collections.Collections.ElementAt\n */
public actual fun CharArray.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}
```kotlin
override val size: Int = this@asList.size
override fun isEmpty(): Boolean = this@asList.isEmpty()
override fun contains(element: Char): Boolean = this@asList.contains(element)
override fun get(index: Int): Char {
    AbstractList.checkElementIndex(index, size)
    return this@asList[index]
}
override fun indexOf(element: Char): Int {
    if ((element as Any?) !is Char) return -1
    return this@asList.indexOf(element)
}
override fun lastIndexOf(element: Char): Int {
    if ((element as Any?) !is Char) return -1
    return this@asList.lastIndexOf(element)
}

/**
 * Returns a 
**
 * List**
 * that wraps the original array.
 * @return

 */
public actual fun CharArray.asList(): List<Char> {
    return this.unsafeCast<Array<Char>>().asList()
}

/**
 * Returns a 
**
 * List**
 * that wraps the original array.
 * @return

 */
@kotlin.internal.InlineOnly
public actual inline fun BooleanArray.asList(): List<Boolean> {
    return this.unsafeCast<Array<Boolean>>().asList()
}

/**
 * Returns a 
**
 * List**
 * that wraps the original array.
 * @return

 */
@SinceKotlin("1.1")
@library("arrayDeepEquals")
public actual infix fun <T> Array<out T>.contentDeepEquals(other: Array<out T>): Boolean {
    definedExternally
}

/**
 * Returns a hash code
 * based on the contents of this array as if it is a List.
 * Nested arrays are treated as lists too.
 * If any of arrays contains itself on any nesting level the behavior is undefined.
 * The elements of other types are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayDeepHashCode")
public actual fun <T> Array<out T>.contentDeepHashCode(): Int {
    definedExternally
}

/**
 * Returns a string representation of the contents of this array as if it is a List.
 * Nested arrays are treated as lists too.
 * If any of arrays contains itself on any nesting level the behavior is undefined.
 */
@SinceKotlin("1.1")
@library("arrayDeepToString")
public actual fun <T> Array<out T>.contentDeepToString(): String {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun <T> Array<out T>.contentEquals(other: Array<out T>): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun ByteArray.contentEquals(other: ByteArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun ShortArray.contentEquals(other: ShortArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun IntArray.contentEquals(other: IntArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun LongArray.contentEquals(other: LongArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun DoubleArray.contentEquals(other: DoubleArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun FloatArray.contentEquals(other: FloatArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun CharArray.contentEquals(other: CharArray): Boolean {
    definedExternally
}

/**
 * Returns `true` if the two specified arrays
 * are *structurally* equal to one another,
 * i.e. contain the same number of the same elements in the same order.
 * The elements are compared for equality with the [equals][Any.equals] function.
 * For floating point numbers it means that ‘NaN’ is equal to itself and ‘-0.0’ is not equal to ‘0.0’.
 */
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun BooleanArray.contentEquals(other: BooleanArray): Boolean {
    definedExternally
}
```

"NaN" is equal to itself and `-0.0` is not equal to `0.0`.

```kotlin
@SinceKotlin("1.1")
@library("arrayEquals")
public actual infix fun LongArray.contentEquals(other: LongArray): Boolean {
    definedExternally
} *structurally* equal to one another, \n * i.e. contain the same number of the same elements in the same order.

* The elements are compared for equality with the \[equals\][Any.equals] function. \n * For floating point numbers it means that 'NaN' is equal to itself and '-0.0' is not equal to '0.0'.

```
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun ShortArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun IntArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun LongArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun FloatArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun DoubleArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun BooleanArray.contentToString(): String {
    definedExternally
}

/**
 * Returns a string representation of the contents of the specified array as if it is
 * 
 * @sample samples.collections.Arrays.ContentOperations.contentToString
 */
@SinceKotlin("1.1")
@library("arrayToString")
public actual fun CharArray.contentToString(): String {
    definedExternally
}

/**
 * Copies this array or its subrange into the [destination] array and returns that
 * 
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun <T> Array<out T>.copyInto(destination: Array<T>, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): Array<T> {
    arrayCopy(this, destination, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * 
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it
 * overlaps with the destination range. 
 * 
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startStartIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * 
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * 
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ByteArray.copyInto(destination: ByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): ByteArray {
    arrayCopy(this.unsafeCast<Array<Byte>>(), destination.unsafeCast<Array<Byte>>(), destinationOffset, startIndex, endIndex)
    return destination
}
same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @param destination the array to copy to.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndexOf] or [endIndex] is out of range of this array indices or when `startIndexOf > endIndex`. It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndexOf] or [endIndex] is out of range of this array indices or when `startIndexOf > endIndex`.

* @throws_indexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

*/

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun ShortArray.copyInto(destination: ShortArray, 
    destinationOffset: Int = 0, 
    startIndex: Int = 0, 
    endIndex: Int = size): ShortArray {
    arrayCopy(this.unsafeCast<Array<Short>>(), 
        destination.unsafeCast<Array<Short>>(), 
        destinationOffset, startIndex, endIndex)
    return destination
}

/**

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @param destination the array to copy to.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndexOf] or [endIndex] is out of range of this array indices or when `startIndexOf > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

*/

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun IntArray.copyInto(destination: IntArray, 
    destinationOffset: Int = 0, 
    startIndex: Int = 0, 
    endIndex: Int = size): IntArray {
    arrayCopy(this.unsafeCast<Array<Int>>(), 
        destination.unsafeCast<Array<Int>>(), 
        destinationOffset, startIndex, endIndex)
    return destination
}

/**

* Copies this array or its subrange into the [destination] array and returns that array.

* It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.

* @param destination the array to copy to.

* @param destinationOffset the position in the [destination] array to copy to, 0 by default.

* @param startIndexOf the beginning (inclusive) of the subrange to copy, 0 by default.

* @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.

* @throws IndexOutOfBoundsException or [IllegalArgumentOutOfRangeException] when [startIndexOf] or [endIndex] is out of range of this array indices or when `startIndexOf > endIndex`.

* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset], or when that index is out of the [destination] array indices range.

*/

@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun LongArray.copyInto(destination: LongArray, 
    destinationOffset: Int = 0, 
    startIndex: Int = 0, 
    endIndex: Int = size): LongArray {
    arrayCopy(this.unsafeCast<Array<Long>>(), 
        destination.unsafeCast<Array<Long>>(), 
        destinationOffset, startIndex, endIndex)
    return destination
}
public actual inline fun FloatArray.copyInto(destination: FloatArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): FloatArray {
    arrayCopy(this.unsafeCast<Array<Float>>(), destination.unsafeCast<Array<Float>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun DoubleArray.copyInto(destination: DoubleArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): DoubleArray {
    arrayCopy(this.unsafeCast<Array<Double>>(), destination.unsafeCast<Array<Double>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun BooleanArray.copyInto(destination: BooleanArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): BooleanArray {
    arrayCopy(this.unsafeCast<Array<Boolean>>(), destination.unsafeCast<Array<Boolean>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the destination range.
 * @param destination the array to copy to.
 * @param destinationOffset the position in the [destination] array to copy to, 0 by default.
 * @param startIndex the beginning (inclusive) of the subrange to copy, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, size of this array by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual inline fun CharArray.copyInto(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): CharArray {
    arrayCopy(this.unsafeCast<Array<Char>>(), destination.unsafeCast<Array<Char>>(), destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual inline fun <T> Array<out T>.copyOf(): Array<T> {
    return this.asDynamic().slice()
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
@Suppress("NOTHING_TO_INLINE")
public
actual inline fun ByteArray.copyOf(): ByteArray {
    return this.asDynamic().slice()
array is truncated to the [newSize].
 * - If [newSize] is greater than the size of the original array, the extra elements in the copy array are filled with zero values.
 * @sample

Fun `copyOf almaFloatArray`:

```kotlin
fun FloatArray.copyOf(newSize: Int): FloatArray {
    require(newSize >= 0) { "Invalid new array size: \$newSize." }
    return fillFrom(this, FloatArray(newSize))
}
```

Fun `copyOf almadoubleArray`:

```kotlin
fun DoubleArray.copyOf(newSize: Int): DoubleArray {
    require(newSize >= 0) { "Invalid new array size: \$newSize." }
    return fillFrom(this, DoubleArray(newSize))
}
```

Fun `copyOf alumbooleanArray`:

```kotlin
fun BooleanArray.copyOf(newSize: Int): BooleanArray {
    require(newSize >= 0) { "Invalid new array size: \$newSize." }
    return withType("BooleanArray", arrayCopyResize(this, newSize, false))
}
```

Fun `copyOf alumcharArray`:

```kotlin
fun CharArray.copyOf(newSize: Int): CharArray {
    require(newSize >= 0) { "Invalid new array size: \$newSize." }
    return withType("CharArray", fillFrom(this, CharArray(newSize)))
}
```

Fun `copyOf alcarray`:

```kotlin
fun <T> Array<out T>.copyOf(newSize: Int): Array<T?> {
    require(newSize >= 0) { "Invalid new array size: \$newSize." }
    return arrayCopyResize(this,
```
@param toIndex the end of the range (exclusive), must be in `fromIndex..<array.size`

/*public actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {\n    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return this.asDynamic().slice(fromIndex, toIndex)\n}*/

public actual fun IntArray.copyOfRange(fromIndex: Int, toIndex: Int): IntArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return this.asDynamic().slice(fromIndex, toIndex)\n}*/

public actual fun LongArray.copyOfRange(fromIndex: Int, toIndex: Int): LongArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return withType("LongArray",\n        this.asDynamic().slice(fromIndex, toIndex))\n}*/

public actual fun Array.copyOfRange(fromIndex: Int, toIndex: Int): Array {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return this.asDynamic().slice(fromIndex, toIndex)\n}*/

public actual fun A copyOfRange(fromIndex: Int, toIndex: Int): CharArray {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    return withType("CharArray",\n        this.asDynamic().slice(fromIndex, toIndex))\n}*/

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun <T> Array<T>.fill(element: T, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}*/

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}*/

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}*/

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public actual fun ByteArray.fill(element: Byte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)\n    this.asDynamic().fill(element, fromIndex, toIndex)\n}*/

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun ShortArray.fill(element: Short, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @param throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @param throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun IntArray.fill(element: Int, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @param throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @param throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun LongArray.fill(element: Long, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @param throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @param throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun FloatArray.fill(element: Float, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @param throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @param throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun DoubleArray.fill(element: Double, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @param throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @param throws IllegalArgumentException if [fromIndex] is greater than [toIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun BooleanArray.fill(element: Boolean, fromIndex: Int = 0, toIndex: Int = size): Unit {
    AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
    this.asDynamic().fill(element, fromIndex, toIndex)
}

*/
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
public
actual fun CharArray.fill(element: Char, fromIndex: Int = 0, toIndex: Int = size): Unit {
AbstractList.checkRangeIndexes(fromIndex, toIndex, size)
this.asDynamic().fill(element, fromIndex, toIndex)
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
fun <T> Array<out T>.plus(element: T): Array<T> {
return
this.asDynamic().concat(arrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun ByteArray.plus(element: Byte): ByteArray {
return
plus(byteArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun ShortArray.plus(element: Short): ShortArray {
return
plus(shortArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun IntArray.plus(element: Int): IntArray {
return
plus(intArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun LongArray.plus(element: Long): LongArray {
return
plus(longArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun FloatArray.plus(element: Float): FloatArray {
return
plus(floatArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun DoubleArray.plus(element: Double): DoubleArray {
return
plus(doubleArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then the given [element].
 */
@Suppress("NOTHING_TO_INLINE")
fun BooleanArray.plus(element: Boolean): BooleanArray {
return
plus(booleanArrayOf(element))
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
@Suppress("ACTUAL_WITHOUT_EXPECT")
fun <T> Array<out T>.plus(elements: Collection<T>): Array<T> {
return
arrayPlusCollection(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun ByteArray.plus(elements: Collection<Byte>): ByteArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun ShortArray.plus(elements: Collection<Short>): ShortArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun IntArray.plus(elements: Collection<Int>): IntArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun LongArray.plus(elements: Collection<Long>): LongArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun FloatArray.plus(elements: Collection<Float>): FloatArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun DoubleArray.plus(elements: Collection<Double>): DoubleArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] collection.
 */
fun BooleanArray.plus(elements: Collection<Boolean>): BooleanArray {
return
fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}
arrayPlusCollection(this, elements)

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual operator fun CharArray.plus(elements: Collection<Char>): CharArray {
    return fillFromCollection(this.copyOf(size + elements.size), this.size, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
public actual inline operator fun <T> Array<out T>.plus(elements: Array<out T>): Array<T> {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ByteArray.plus(elements: ByteArray): ByteArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun ShortArray.plus(elements: ShortArray): ShortArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun IntArray.plus(elements: IntArray): IntArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun LongArray.plus(elements: LongArray): LongArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun FloatArray.plus(elements: FloatArray): FloatArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun DoubleArray.plus(elements: DoubleArray): DoubleArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun BooleanArray.plus(elements: BooleanArray): BooleanArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@Suppress("NOTHING_TO_INLINE")
public actual inline operator fun CharArray.plus(elements: CharArray): CharArray {
    return primitiveArrayConcat(this, elements)
}

/**
 * Returns an array containing all elements of the original array and then all the given [element].
 */
@Suppress("ACTUAL_WITHOUT_EXPECT", "NOTHING_TO_INLINE")
public actual inline fun <T> Array<out T>.plusElement(element: T): Array<T> {
    return this.asDynamic().concat(arrayOf(element))
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun IntArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun LongArray.sort(): Unit {
    if (size > 1) sort {
        a: Long, b: Long -> a.compareTo(b)
    }
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun ByteArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun ShortArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun FloatArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun DoubleArray.sort(): Unit {
    definedExternally
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@library("primitiveArraySort")
public actual fun FloatArray.sort(): Unit {
    definedExternally
}
CharArray.sort(): Unit {\n    definedExternally\n} \n\nVarious functions for sorting arrays in-place according to natural order of elements. The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.\n\nSorts the array in-place according to the order specified by the given [comparison] function.

org.jetbrains.kotlin.js.js("[]").slice.call(this)\n\norg.jetbrains.kotlin.js.js("[]").slice.call(this)\n\norg.jetbrains.kotlin.js.js("[]").slice.call(this)\n\norg.jetbrains.kotlin.js.js("[]").slice.call(this)
**Comparator**

* objects. As soon as the [Comparator] instances returned by a function for `a` and `b` values do not compare as equal, the result of that comparison is returned.

```kotlin
private fun <T> compareValuesByImpl(a: T, b: T, selectors: Array<out (T) -> Comparable<*>?>): Int {
    for (fn in selectors) {
        val v1 = fn(a)
        val v2 = fn(b)
        val diff = compareValues(v1, v2)
        if (diff != 0) return diff
    }
    return 0
}
```

```kotlin
fun <T : Comparable<*>> compareValues(a: T?, b: T?): Int {
    if (a === b) return 0
    if (a == null) return -1
    if (b == null) return 1

    @Suppress("UNCHECKED_CAST")
    return (a as Comparable<Any>).compareTo(b)
}
```

### Examples

```kotlin
// Samples
samples.comparisons.Comparisons.compareValuesBy

@Suppress("NOTHING_TO_INLINE")
public fun <T> compareValuesBy(a: T, b: T, vararg selectors: (T) -> Comparable<*>) ?: Int {
    require(selectors.size > 0)
    return compareValuesByImpl(a, b, selectors)
}
```

```kotlin
fun <T> compareBy(vararg selectors: (T) -> Comparable<*>?): Comparator<T> {
    require(selectors.size > 0)
    return Comparator { a, b ->
        compareValuesByImpl(a, b, selectors)
    }
}
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> compareBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(a, b, selector)
    }
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T, K> compareBy(comparator: Comparator<in K>, crossinline selector: (T) -> K): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(a, b, comparator, selector)
    }
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T> compareByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =
    Comparator { a, b ->
        compareValuesBy(b, a, selector)
    }
```

```kotlin
@kotlin.internal.InlineOnly
public inline fun <T, K> compareByDescending(comparator: Comparator<in K>,

@Suppress("NOTHING_TO_INLINE")
public fun <T> compareValuesByDescending(crossinline selector: (T) -> Comparable<*>) ?: Int {
    return compareValuesByDescendingImpl(a, b, selectors)
}
```

```kotlin
fun <T> compareByDescendingWithSingleSelector

@Suppress("NOTHING_TO_INLINE")
public fun <T> compareByDescendingWithSingleSelector
```
crossinline selector: (T) -> K): Comparator<T> =\n Comparator { a, b -> compareValuesBy(b, a, comparator,
selector) }\n\n/**\n * Creates a comparator comparing values after the primary comparator defined them equal. It
uses\n * the function to transform value to a [Comparable] instance for comparison.\n *\n * @sample
samples.comparisons.Comparisons.thenBy\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T>
Comparator<T>.thenBy(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n Comparator { a, b >\n
val previousCompare = this@thenBy.compare(a, b)\n
if (previousCompare != 0) previousCompare else
compareValuesBy(a, b, selector)\n }\n\n/**\n * Creates a comparator comparing values after the primary
comparator defined them equal. It uses\n * the [selector] function to transform values and then compares them with
the given [comparator].\n *\n * @sample samples.comparisons.Comparisons.thenByWithComparator\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <T, K> Comparator<T>.thenBy(comparator: Comparator<in K>,
crossinline selector: (T) -> K): Comparator<T> =\n Comparator { a, b ->\n
val previousCompare =
this@thenBy.compare(a, b)\n
if (previousCompare != 0) previousCompare else compareValuesBy(a, b,
comparator, selector)\n }\n\n/**\n * Creates a descending comparator using the primary comparator and\n * the
function to transform value to a [Comparable] instance for comparison.\n *\n * @sample
samples.comparisons.Comparisons.thenByDescending\n */\n@kotlin.internal.InlineOnly\npublic inline fun <T>
Comparator<T>.thenByDescending(crossinline selector: (T) -> Comparable<*>?): Comparator<T> =\n
Comparator { a, b ->\n
val previousCompare = this@thenByDescending.compare(a, b)\n
if
(previousCompare != 0) previousCompare else compareValuesBy(b, a, selector)\n }\n\n/**\n * Creates a
descending comparator comparing values after the primary comparator defined them equal. It uses\n * the [selector]
function to transform values and then compares them with the given [comparator].\n *\n * @sample
samples.comparisons.Comparisons.thenByDescendingWithComparator\n */\n@kotlin.internal.InlineOnly\npublic
inline fun <T, K> Comparator<T>.thenByDescending(comparator: Comparator<in K>, crossinline selector: (T) ->
K): Comparator<T> =\n Comparator { a, b ->\n
val previousCompare = this@thenByDescending.compare(a,
b)\n
if (previousCompare != 0) previousCompare else compareValuesBy(b, a, comparator, selector)\n
}\n\n\n/**\n * Creates a comparator using the primary comparator and function to calculate a result of comparison.\n
*\n * @sample samples.comparisons.Comparisons.thenComparator\n */\n@kotlin.internal.InlineOnly\npublic inline
fun <T> Comparator<T>.thenComparator(crossinline comparison: (a: T, b: T) -> Int): Comparator<T> =\n
Comparator { a, b ->\n
val previousCompare = this@thenComparator.compare(a, b)\n
if (previousCompare
!= 0) previousCompare else comparison(a, b)\n }\n\n/**\n * Combines this comparator and the given [comparator]
such that the latter is applied only\n * when the former considered values equal.\n *\n * @sample
samples.comparisons.Comparisons.then\n */\npublic infix fun <T> Comparator<T>.then(comparator:
Comparator<in T>): Comparator<T> =\n Comparator { a, b ->\n
val previousCompare =
this@then.compare(a, b)\n
if (previousCompare != 0) previousCompare else comparator.compare(a, b)\n
}\n\n/**\n * Combines this comparator and the given [comparator] such that the latter is applied only\n * when the
former considered values equal.\n *\n * @sample samples.comparisons.Comparisons.thenDescending\n */\npublic
infix fun <T> Comparator<T>.thenDescending(comparator: Comparator<in T>): Comparator<T> =\n
Comparator<T> { a, b ->\n
val previousCompare = this@thenDescending.compare(a, b)\n
if
(previousCompare != 0) previousCompare else comparator.compare(b, a)\n }\n\n// Not so useful without type
inference for receiver of expression\n/**\n * Extends the given [comparator] of non-nullable values to a comparator
of nullable values\n * considering `null` value less than any other value.\n *\n * @sample
samples.comparisons.Comparisons.nullsFirstLastWithComparator\n */\npublic fun <T : Any>
nullsFirst(comparator: Comparator<in T>): Comparator<T?> =\n Comparator { a, b ->\n
when {\n
a
=== b -> 0\n
a == null -> -1\n
b == null -> 1\n
else -> comparator.compare(a, b)\n
}\n
}\n\n/**\n * Provides a comparator of nullable [Comparable] values\n * considering `null` value less than any other
value.\n *\n * @sample samples.comparisons.Comparisons.nullsFirstLastComparator\n
*/\n@kotlin.internal.InlineOnly\npublic inline fun <T : Comparable<T>> nullsFirst(): Comparator<T?> =
nullsFirst(naturalOrder())\n\n/**\n * Extends the given [comparator] of non-nullable values to a comparator of
nullable values\n * considering `null` value greater than any other value.\n *\n * @sample

Open Source Used In IOx-FogDirector-AC18 1.17.0 2160


samples.comparisons.Comparisons.nullsFirstLastWithComparator

@Suppress("UNCHECKED_CAST")
public fun <T : Any>
    nullsLast(comparator: Comparator<in T>): Comparator<T?> =
        Comparator { a, b ->
            when {
                a === b -> 0
                a == null -> 1
                b == null -> -1
                else -> comparator.compare(a, b)
            }
        }

/**
 * Provides a comparator of nullable [Comparable] values*\n * considering `null` value greater than any
 * other value.*\n */
public fun <T : Comparable<T>>
    nullsLast(): Comparator<T?> =
        nullsLast(naturalOrder())

/**
 * Returns a comparator that imposes the reverse ordering
 * of this comparator.*\n */
private class ReversedComparator<T>(public val comparator: Comparator<T>) : Comparator<T> {
    override fun compare(a: T, b: T): Int = comparator.compare(b, a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<T> = comparator
}

private object NaturalOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = a.compareTo(b)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = ReverseOrderComparator
}

private object ReverseOrderComparator : Comparator<Comparable<Any>> {
    override fun compare(a: Comparable<Any>, b: Comparable<Any>): Int = b.compareTo(a)
    @Suppress("VIRTUAL_MEMBER_HIDDEN")
    fun reversed(): Comparator<Comparable<Any>> = NaturalOrderComparator
}

/**
 * Always throws [NotImplementedError] stating that operation is not implemented.
 * @param reason a string
 */
@kotlin.internal.InlineOnly
public inline fun TODO(reason: String): Nothing = throw NotImplementedError("An operation is not implemented: $reason")

/**
 * Calls the specified function [block] and returns its result.*\n */
@kotlin.internal.InlineOnly
public inline fun <R> run(block: () -> R): R {
    contract {
        callsInPlace(block, InvocationKind.EXACTLY_ONCE)
    }
    return block()
}

/**
 * Calls
 * the specified function [block] with `this` value as its receiver and returns its result.*\n */
public fun <T>
    callInPlace(block: InvocationKind, EXACTLY_ONCE) {
        return block()}

/**
 * An exception is thrown to indicate that a method body remains to be implemented.*\n */
public class NotImplementedError(message: String = "An operation is not implemented.") : Error(message)

/**
 * Always throws [NotImplementedError] stating that operation is not implemented.*\n */
public class NotImplementedError(message: String) : Error(message)
```kotlin
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.js.*

/**
 * Returns the greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public actual fun <T : Comparable<T>> maxOf(a: T, b: T): T {
    return if (a >= b) a else b
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Byte, b: Byte): Byte {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Byte>()
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Short, b: Short): Short {
    return Math.max(a.toInt(), b.toInt()).unsafeCast<Short>()
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun maxOf(a: Int, b: Int): Int {
    return Math.max(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("NOTHING_TO_INLINE")
public actual inline fun maxOf(a: Long, b: Long): Long {
    return if (a >= b) a else b
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2162
fun maxOf(a: Float, b: Float): Float {
    return Math.max(a, b)
}

/**
 * Returns the greater of two values.
 */
fun maxOf(a: Double, b: Double): Double {
    return Math.max(a, b)
}

fun maxOf(a: Int, b: Int, c: Int): Int {
    return Math.max(maxOf(a, b), c)
}

fun maxOf(a: Long, b: Long, c: Long): Long {
    return maxOf(a, maxOf(b, c))
}

fun maxOf(a: Float, b: Float, c: Float): Float {
    return Math.max(a, b, c)
}

fun minOf(a: Byte, b: Byte): Byte {
    return Math.min(a, b)
}

fun minOf(a: Short, b: Short): Short {
    return Math.min(a, b)
}

fun minOf(a: Int, b: Int, c: Int): Int {
    return Math.min(a, b, c)
}

fun minOf(a: Long, b: Long): Long {
    return if (a <= b) a else b
}

fun minOf(a: Float, b: Float): Float {
    return Math.min(a, b)
}

fun minOf(a: Double, b: Double): Double {
    return Math.min(a, b)
}

fun minOf(a: Byte, b: Byte, c: Byte): Byte {
    return Math.min(a, b, c)
}

fun minOf(a: Short, b: Short, c: Short): Short {
    return Math.min(a, b, c)
}

fun minOf(a: Int, b: Int, c: Int): Int {
    return Math.min(a, b, c)
}

fun minOf(a: Long, b: Long, c: Long): Long {
    return if (a <= b) a else b
}

fun minOf(a: Float, b: Float, c: Float): Float {
    return Math.min(a, b, c)
}

fun minOf(a: Double, b: Double, c: Double): Double {
    return Math.min(a, b, c)
}
c.toInt()).unsafeCast<Short>()
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
@Suppress("DEPRECATION_ERROR")
public actual inline fun minOf(a: Int, b: Int, c: Int): Int {
    return Math.min(a, b)
}
Iterable<T>.elementAtOrNull(index: Int): T? {
    if (this is List)
        return this.getOrNull(index)
    if (index < 0)
        return null
    val iterator = iterator() 
    var count = 0
    while (iterator.hasNext()) {
        val element = iterator.next()
        if (index == count++)
            return element
    }
    return null
}

Iterable<T>.first(): T {
    when (this) {
        is List -> return this.first()
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                throw NoSuchElementException("Collection is empty.")
            return iterator.next()
        }
    }
}

List<T>.first(): T {
    if (isEmpty())
        throw NoSuchElementException("List is empty.")
    return this[0]
}

List<T>.indexOf(element: T): Int {
    if (this is List) return this.indexOf(element)
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            return index
        index++
    }
    return -1
}

List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}

List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var index = lastIndex
    while (index > 0 & & index <= lastIndex)
        index--
    for (item in this) {
        if (index == 0 & & index <= lastIndex)
            return index
        index--
    }
    return -1
}

List<T>.getOrDefault(index: Int, defaultValue: (Int) -> T): T {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

List<T>.getOrNull(index: Int): T? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

List<T>.find(predicate: (T) -> Boolean): T? {
    var index = 0
    while (index <= lastIndex)
        if (predicate(this[index]))
            return this[index]
        index++
    return null
}

List<T>.findLast(predicate: (T) -> Boolean): T? {
    var index = lastIndex
    while (index >= 0 & & index >= lastIndex)
        index--
    for (item in this) {
        if (index == 0 & & index >= lastIndex)
            return item
        index--
    }
    return null
}

List<T>.findLast(predicate: (T) -> Boolean): T? {
    var index = lastIndex
    while (index >= 0 & & index >= lastIndex)
        index--
    for (item in this) {
        if (index == 0 & & index >= lastIndex)
            return item
        index--
    }
    return null
}

List<T>.indexOf(element: T): Int {
    return if (this is List) this.indexOf(element) else {
        var index = 0
        while (index <= lastIndex)
            if (index == 0 & & index <= lastIndex)
                return index
            index++
        return -1
    }
}

List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    return if (this is List) this.indexOfFirst(predicate) else {
        var index = 0
        while (index <= lastIndex)
            if (index == 0 & & index <= lastIndex)
                return index
            index++
        return -1
    }
}

List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    return if (this is List) this.indexOfLast(predicate) else {
        var index = lastIndex
        while (index >= 0 & & index >= lastIndex)
            index--
        for (item in this) {
            if (index == 0 & & index >= lastIndex)
                return item
            index--
        }
        return -1
    }
}
@SinceKotlin("1.3")
public fun <T> Collection<T>.random(random: Random): T {
    if (isEmpty())
        throw NoSuchElementException("List contains no element matching the given predicate.")
    @Suppress("UNCHECKED_CAST")
    return last as T
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the collection is empty.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.last(): T {
    when (this) {
        is List -> return if (isEmpty()) null else this[size - 1]
        else -> {
            val iterator = iterator()
            if (!iterator.hasNext())
                return null
            var last = iterator.next()
            while (iterator.hasNext())
                last = iterator.next()
            return last
        }
    }
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <T> List<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns last index of the last element matching the given [predicate], or -1 if the list does not contain such element.
 */
public fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns index of the last element matching the [predicate], or -1 if the list does not contain such element.
 */
public fun <T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns the last element matching the given [predicate], or -1 if the list does not contain such element.
 */
public fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of the last element matching the given [predicate], or -1 if the list does not contain such element.
 */
public fun <T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <T> List<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns last index of [element], or -1 if the list does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Iterable<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}
throw NoSuchElementException("Collection is empty.")\nreturn elementAt(random.nextInt(size))\n\n// Returns a random element from this collection, or null if this collection is empty.\n\n@SinceKotlin("1.3")@ExperimentalStdlibApi\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.randomOrNull(): T? {\n    if (isEmpty()) return null\n    val element = elementAt(random.nextInt(size))\n    return element\n}\n
/**\n * Returns a random element from this collection, or throws an exception if the collection is empty or has more than one element.\n */\n@SinceKotlin("1.3")\npublic fun <T> Collection<T>.randomOrNull(random: Random): T? {\n    if (isEmpty()) return null\n    return elementAt(random.nextInt(size))\n}\n
/**\n * Returns the single element, or throws an exception if the list is empty or has more than one element.\n */\n\n@SinceKotlin("1.3")\npublic fun <T> List<T>.single(): T {\n    return when (size) {\n        0 -> throw NoSuchElementException("List is empty.")\n        1 -> this[0]\n        else -> throw IllegalArgumentException("Collection has more than one element.")\n    }\n}\n
/**\n * Returns the single element matching the given [predicate], or throws exception if there is no or more than one matching element.\n */\npublic inline fun <T> Iterable<T>.singleOrNull(predicate: (T) -> Boolean): T? {\n    var single: T? = null\n    var found = false\n    for (element in this) {\n        if (predicate(element)) {\n            if (found) throw IllegalArgumentException("Collection contains more than one matching element.")\n            single = element\n            found = true\n        }\n    }\n    if (!found) throw NoSuchElementException("Collection contains no element matching the predicate.")\n    return single\n}\n
/**\n * Returns a list containing all elements except first [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n */\npublic fun <T> List<T>.drop(n: Int): List<T> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return this\n    val list: ArrayList<T> = if (this is List<*>) {\n        val resultSize = size - n\n        if (resultSize <= 0) return emptyList()\n        if (resultSize == 1) return listOf(last())\n        list = ArrayList<T>(resultSize)\n        if (this is RandomAccess) {\n            for (index in n until size)\n                list.add(this[index])\n        } else {\n            for (item in listIterator(n))\n                list.add(item)\n        }\n        list\n    } else {\n        list = ArrayList<T>()\n    }\n    var count = 0\n    for (item in this) {\n        if (count >= n) list.add(item) else ++count\n    }\n    return list.optimizeReadOnlyList()\n}\n
/**\n * Returns a list containing all elements except last [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n */\npublic fun <T> List<T>.dropLast(n: Int): List<T> {\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    return take((size - n).coerceAtLeast(0))\n}\n
/**\n * Returns a list containing all elements except last elements that satisfy the given [predicate].\n */\npublic inline fun <T> List<T>.dropLastWhile(predicate: (T) -> Boolean): List<T> {\n    if (!isEmpty()) {\n        val iterator = iterator()\n        while (iterator.hasNext()) {\n            val element = iterator.next()\n            if (iterator.hasNext()) return null\n            val single = iterator.next()\n            if (iterator.hasNext()) return null\n            if (predicate(element)) {\n                if (found) return null\n                single = element\n                found = true\n            }\n        }\n        if (!found) return null\n        return single\n    }\n}
take(iterator.nextIndex() + 1) \n } \n } return emptyList() \n * Returns a list containing all elements except first elements that satisfy the given [predicate]. \n * @param \n * \n return listOf(\n ) \n * Returns a list containing only elements matching the given [predicate]. \n * @param \n * @sample \n take(iterator.nextIndex() + 1) \n } \n } return listOf(\n ) \n * Returns a list containing all elements matching the given [predicate]. \n * @param \n * @sample

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.dropWhile
 */
public inline fun <T> List<T>.dropWhile(predicate: (T) -> Boolean): List<T> { \n var yielding = false \n val list = ArrayList<T>() \n for (item in this) \n if (yielding) \n list.add(item) \n else if (predicate(item)) { \n list.add(item) \n yielding = true \n } \n return list() \n * Returns a list containing only elements matching the given [predicate].
 * @param

/**
 * Returns a list containing all elements except first elements that satisfy the given 
 * [predicate].
 * 
 * @sample
 */
public inline fun <T> Iterable<T>.filter(predicate: (T) -> Boolean): List<T> { \n return filterTo(ArrayList<T>(), predicate) \n * Appends all elements matching the given [predicate] to the given [destination].
 * @param

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 * @param

/**
 * Returns a list containing all elements that are instances of specified type parameter R.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterIsInstance(): List<R> \n * Returns a list containing all elements that are not matching the given [predicate].
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterNot(): List<T> \n * Returns a list containing all elements that are null.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterNotNull(): List<T> \n * Returns a list containing all elements that are not null.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterNotNull(): List<T> \n * Returns a list containing all elements that are not null.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterNotLast(): List<T> \n * Returns a list containing all elements that are not last.
 * 
 * @throws IllegalArgumentException if [n] is negative.
 * 
 * @sample
 */
public inline fun <T> List<T>.filterNotLast(): List<T> \n * Returns a list containing all elements that are not last.

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: Iterable<Int>): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
 * 
 * @sample
 */
public fun <T> List<T>.slice(indices: IntRange): List<T> { \n return list() \n * Returns a list containing elements at indices in the specified [indices] range.
 * 
 * @throws IllegalArgumentException if [indices] range is empty.
RandomAccess) {
    for (index in size - n until size)
        list.add(this[index])
} else {
    for (item in list)
        list.add(item)
}

return list
}

/**
 * Returns a list containing last elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
fun <T> List<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {
    if (isEmpty())
        return emptyList()
    val iterator = listIterator(size)
    while (iterator.hasNext()) {
        if (!predicate(iterator.next()))
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
        else {
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
            for (item in listIterator(size - n))
                list.add(item)
        }
    }
    return list
}

/**
 * Returns a list containing first elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
fun <T> List<T>.takeFirstWhile(predicate: (T) -> Boolean): List<T> {
    if (isEmpty())
        return emptyList()
    val iterator = listIterator(size)
    while (iterator.hasNext()) {
        if (predicate(iterator.next()))
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
            for (item in listIterator(size - n))
                list.add(item)
        else {
            iterator.next()
            val expectedSize = size - iterator.nextIndex()
            for (item in listIterator(size - n))
                list.add(item)
        }
    }
    return list
}

/**
 * Reverses elements in the list in-place.
 */
fun <T> MutableList<T>.reverse(): Unit

/**
 * Returns a list with elements in reversed order.
 */
fun <T> List<T>.reversed(): List<T> {
    if (this is Collection && size <= 1)
        return toList()
    return ArrayList<T>(size).apply {
        @Suppress("UNCHECKED_CAST")
        val array = (toList() as Array<T>).toTypedArray()
        array.sort() as List<T>
    }
}

/**
 * Sorts elements in the list in-place according to
 * natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other.
 * Returns a list of all elements sorted according to their natural sort order.
 */
fun <T, R : Comparable<R>> MutableList<T>.sortBy(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareBy(selector))
}

/**
 * Sorts elements in the list in-place descending according to
 * natural sort order of the value returned by specified [selector] function.
 * The sort is _stable_. It means that equal elements preserve their order relative to each other.
 * Returns a list of all elements sorted according to their natural sort order.
 */
fun <T, R : Comparable<R>> MutableList<T>.sortByDescending(crossinline selector: (T) -> R?): Unit {
    if (size > 1)
        sortWith(compareByDescending(selector))
}

/**
 * Sorts elements in the list in-place descending according to their natural sort order.
 */
fun <T : Comparable<T>> MutableList<T>.sortDescending(): Unit {
    sortWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to their natural sort order.
 */
fun <T : Comparable<T>> Collection<T>.sortedDescending(): List<T> {
    return sortedWith(reverseOrder())
}

/**
 * Returns a list of all elements sorted according to the specified [comparator].
 * The sort is _stable_. It means that equal elements preserve their order relative to each other.
 * Returns a list of all elements sorted according to the specified [comparator].
 */
fun <T> Collection<T>.sortedWith(comparator: Comparator<T>): List<T> {
    if (this is Collection)
        return toList() @Suppress("UNCHECKED_CAST")
    return (toList() as Array<T>).apply {
        sort() } as List<T>
}

/**
 * Returns a list containing last elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
fun <T> Iterable<T>.takeLastWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this)
        if (!predicate(item))
            break
        list.add(item)
    return list
}

/**
 * Returns a list containing first elements
 * satisfying the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.take
 */
fun <T> Iterable<T>.takeFirstWhile(predicate: (T) -> Boolean): List<T> {
    val list = ArrayList<T>()
    for (item in this)
        if (predicate(item))
            list.add(item)
    return list
}
result

public fun Collection<Byte>.toByteArray(): ByteArray {
    val result = ByteArray(size)
    var index = 0
    for (element in this)
        result[index++]

public fun Collection<Double>.toDoubleArray(): DoubleArray {
    val result = DoubleArray(size)
    var index = 0
    for (element in this)
        result[index++]

public fun Collection<Long>.toLongArray(): LongArray {
    val result = LongArray(size)
    var index = 0
    for (element in this)
        result[index++]

public fun Collection<Short>.toShortArray(): ShortArray {
    val result = ShortArray(size)
    var index = 0
    for (element in this)
        result[index++]

public fun Collection<T, K, V>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}

public inline fun <T, K, M : MutableMap<in K, in T>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this)
        destination.put(keySelector(element), element)
    return destination
}

public fun Collection<T, K, V>.associateByToWithValueTransform(transform: (T) -> V): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByToWithValueTransform(LinkedHashMap<K, V>(capacity), transform)
}

public fun Collection<T, K, M :MutableMap<in K, in T>>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this)
        destination.put(keySelector(element), element)
    return destination
}

public fun Collection<T, K, V>.associateByWithValueTransform(): Map<K, V> {
    val capacity = mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)
    return associateByWithValueTransform(LinkedHashMap<K, V>(capacity), transform)
}
V, M : MutableMap<in K, in V>> Iterable<T>.associateByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M { for (element in this) { destination.put(keySelector(element), valueTransform(element)); } return destination } // Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given collection.

V, M : MutableMap<in K, in V>> Iterable<T>.associateTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M { for (element in this) { destination.put(keySelector(element), valueTransform(element)) } return destination } // Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection where keys are elements from the given collection and values are produced by the [valueSelector] function applied to each element.

V, M : Collection<in V, in K>> Iterable<T>.associateWithTo(destination: M, valueSelector: (K) -> V): M { for (element in this) { destination.put(element, valueSelector(element)) } return destination } // Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection where key is the element itself and value is produced by the [valueSelector] function applied to that key.

V, M : Collection<in V, in K>> Iterable<T>.associateWith(valueSelector: (K) -> V): M { val result = LinkedHashMap<K, V>(mapCapacity(collectionSizeOrDefault(10)).coerceAtLeast(16)) return associateWithTo(result, valueSelector) } // Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection where key is the element itself and value is produced by the [valueSelector] function applied to that key.

V, M : Collection<in V, in K>> Iterable<T>.associateWithTo(destination: M, valueSelector: (K) -> V): M { for (element in this) { destination.put(element, valueSelector(element)) } return destination } // Populates and returns the [destination] mutable map with key-value pairs for each element of the given collection where key is the element itself and value is produced by the [valueSelector] function applied to that key.

V, M : Collection<in V, in K>> Iterable<T>.toCollection(destination: C): C { for (item in this) { destination.add(item) } return destination } // Appends all elements to the given [destination] collection.

V, M : Collection<in V, in K>> Iterable<T>.toList(): List<T> { if (this is Collection) { return when (size) { 0 -> emptyList() 1 -> listOf(if (this is List) get(0) else iterator().next()) else -> this.toMutableList().optimizeReadOnlyList() } } return this.toMutableList().optimizeReadOnlyList() } // Returns a [List] containing all elements.

V, M : Collection<in V, in K>> Iterable<T>.toSet(): Set<T> { if (this is Collection) { return when (size) { 0 -> emptySet() 1 -> setOf(if (this is List) this[0] else iterator().next()) else -> toCollection(LinkedHashSet<T>(mapCapacity(size))).optimizeReadOnlySet() } } return toCollection(LinkedHashSet<T>(this)).optimizeReadOnlySet() } // Returns a [Set] of all elements.

V, M : Collection<in V, in K>> Iterable<T>.flatMap(transform: (T) -> Iterable<R>): List<R> { for (item in this) { return flatMapTo(ArrayList<R>(), transform) } } // Returns a [List] of all elements yielded from results of [transform] function being invoked on each element of original collection.

V, M : Collection<in V, in K>> Iterable<T>.flatMapTo(destination: C, transform: (T) -> Iterable<R>): C { for (element in this) { val list = transform(element) destination.addAll(list) } return destination } // Returns a [Map] where keys are elements from the given collection and values are produced by the [keySelector] function applied to each element and returns a map where each group key is associated with a list of corresponding elements.

V, M : Collection<in V, in K>> Iterable<T>.associateTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M { for (element in this) { destination.put(keySelector(element), valueTransform(element)) } return destination } // Populates and returns the [destination] mutable map with key-value pairs provided by [transform] function applied to each element of the given collection.

V, M : Collection<in V, in K>> Iterable<T>.associateByTo(destination: M, transform: (T) -> Pair<K, V>): M { for (element in this) { return destination } // Returns a [Map] where keys are elements from the given collection and values are produced by the [valueSelector] function applied to each element.
Iterable<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to the element
 * and returns a map where each group key is associated with a list of corresponding values.
 * @param [values] are the values returned by the [valueTransform] function applied to each element of the
 * original collection
 * @param [keys] are the keys returned by the [keySelector] function applied to each element of the
 * original collection
 * @return a map where each group key is associated with a list of corresponding values
 */
public inline fun <T, K, M : MutableMap<K, MutableList<T>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}

Iterable<T>.groupBy(keySelector: (T) -> K, valueTransform: (T) -> V): Map<K, List<V>> {
    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original collection
 * by the key returned by the given [keySelector] function applied to each element and
 * puts to the [destination] map each group key associated with a list of corresponding elements.
 * @param [values] are the values returned by the [valueTransform] function applied to each element of the
 * original collection
 * @param [keys] are the keys returned by the given [keySelector] function applied to each element
 * @return a map where each group key is associated with a list of corresponding values
 */
public inline fun <T, K, M : MutableMap<K, MutableList<V>>> Iterable<T>.groupByTo(destination: M, keySelector: (T) -> K, valueTransform: (T) -> V): M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<V>() }
        list.add(valueTransform(element))
    }
    return destination
}

Iterable<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K>{
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

Iterable<T>.map(transform: (T) -> R): List<R> {
    return mapTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)
}

Iterable<T>.mapIndexed(transform: (index: Int, T) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)
}

Iterable<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(), transform)
}

Iterable<T>.mapIndexedNotNullTo(destination: C, keySelector: (T) -> R?): List<R> {
    return mapIndexedNotNullTo(ArrayList<R>(collectionSizeOrDefault(10)), transform)
}

Collections.Map anus */public inline fun <T, K, V> Iterable<T, K, V>
MutableCollection<in R>> Iterable<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R) { C {
    var index = 0
    for (item in this)
        destination.add(transform(checkIndexOverflow(index++), item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given [transform]
 * function to each element in the original collection. *
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapNotNullTo(destination: C, transform: (T) -> R?): List<R> {
    return mapNotNullTo(destination, transform)
}

/**
 * Applies the given [transform] function to each element in the original collection
 * and appends only the non-null results to the given [destination]. *
 */
public inline fun <T, R : Any, C : MutableCollection<in R>> Iterable<T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original collection
 * into an [IndexedValue] containing the index of that element and the element itself. *
 */
public fun <T> Iterable<T>.withIndex(): Iterable<IndexedValue<T>> {
    return IndexingIterable { iterator() }
}

/**
 * Returns a list containing only distinct elements from the given collection. *
 * The elements in the resulting list are in the same order as they were in the source collection. *
 */
public fun <T> Iterable<T>.distinct(): List<T> {
    return toMutableSet().toList()
}

/**
 * Returns a list containing only elements from the given collection
 * that are not contained in the specified collection. *
 * Those elements of the [other] collection that are unique are iterated in
 * the end of the list. *
 */
public infix fun <T> Iterable<T>.intersect(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.retainAll(other)
    return set
}

/**
 * Returns a list containing only elements from the given collection
 * that are not contained in the specified collection. *
 * Those elements of the [other] collection that are unique are iterated in
 * the end of the list. *
 */
public infix fun <T> Iterable<T>.subtract(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.removeAll(other)
    return set
}

/**
 * Returns a mutable set containing all distinct elements from the given collection. *
 * The returned set preserves the element iteration order of the original collection. *
 */
public fun <T> Iterable<T>.toMutableSet(): MutableSet<T> {
    return when (this) {
        is Collection<T> -> LinkedHashSet(this)
        else -> toCollection(LinkedHashSet<T>())
    }
}

/**
 * Returns a list containing all elements that are contained in at least one of these collections use [union]. *
 */
public fun <T, K> Iterable<T>.union(other: Iterable<T>): Set<T> {
    val set = this.toMutableSet()
    set.addAll(other)
    return set
}

/**
 * Returns `true` if all elements match the given [predicate]. *
 */
public inline fun <T> Iterable<T>.all(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return true
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if at least one element matches the given [predicate]. *
 */
public inline fun <T> Iterable<T>.any(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty()) return false
    for (element in this) if (predicate(element)) return true
    return false
}

Open Source Used In IOx-FogDirector-AC18 1.1.70.2 2173
Open Source Used In IOx-FogDirector-AC18 1.17.0  2174

```kotlin
/**
 * Returns the largest element or `null` if there are no elements.
 * If any of elements is `NaN` returns `NaN`.
 */
@SinceKotlin("1.1")
public fun Iterable<Float>.max(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
    }
    return max
}

/**
 * Returns the largest element or `null` if there are no ele-
 * ments.
 */
@SinceKotlin("1.1")
public fun Iterable<Double>.max(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (max < e) max = e
        }n
    }
    return max
}
```

}``
maxValue = selector(maxElem) do { if (maxValue < v) { maxElem = e
        maxValue = v
    }
    val e = iterator.next()
    val v = selector(e) do {
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}

maxValue = selector(maxElem) do { if (maxValue < v) { maxElem = e
        maxValue = v
    }
    val e = iterator.next()
    val v = selector(e) do {
        if (maxValue < v) {
            maxElem = e
            maxValue = v
        }
    } while (iterator.hasNext())
    return maxElem
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun <T> Iterable<T>.min(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
@SinceKotlin("1.1")
public fun <T : Comparable<T>> Iterable<T>.min(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (min > e) min = e
    }
    return min
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
public inline fun <T, R : Comparable<R>> Iterable<T>.minBy(selector: (T) -> R): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    if (!iterator.hasNext()) return minElem
    var minValue = selector(minElem) do { if (minValue < v) { minElem = e
            minValue = v
        }
    } while (iterator.hasNext())
    return minElem
}

/**
 * Returns the smallest element or `null` if there are no elements.
 */
public fun <T> Iterable<T>.minWith(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(min, e) > 0) min = e
    }
    return min
}

/**
 * Returns `true` if the collection has no elements.
 */
public fun <T> Iterable<T>.none(): Boolean {
    if (this is Collection) return isEmpty()
    return !iterator().hasNext()
}

/**
 * Returns `true` if no elements match the given predicate.
 */
public inline fun <T> Iterable<T>.none(predicate: (T) -> Boolean): Boolean {
    if (this is Collection && isEmpty())
        return true
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Performs the given action on each element and returns the collection itself afterwards.
 */
@SinceKotlin("1.1")
public inline fun <T, C : Iterable<T>> C.onEach(action: (T) -> Unit): C {
    return apply { for (element in this)
            action(element) }
}

/**
 * Accumulates value starting with the first element and applying operation to each element.
 */
public inline fun <S, T : S> Iterable<T>.reduce(operation: (acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var accumulator: S = iterator.next() do { while (iterator.hasNext())
            accumulator = operation(accumulator, iterator.next())
        } return accumulator
}

/**
 * Accumulates value starting with the first element and applying operation to each element with its index in the original collection.
 */
public inline fun <S, T : S> Iterable<T>.reduceIndexed(operation: (index: Int, acc: S, T) -> S): S {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty collection can't be reduced.")
    var index = 1
    var accumulator: S = iterator.next() do { while (iterator.hasNext())
            accumulator = operation(accumulator, iterator.next())
        } return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from left to right to the current accumulator value and each element.
 */
public inline fun <S> Collection<S>.accumulateLeft(operation: (acc: S, T) -> S): S {
    var accumulator: S = this.first() do { while (this.hasNext())
            accumulator = operation(accumulator, this.next())
        } return accumulator
}

/**
 * Accumulates value starting with the first element and applying [operation] from right to left to the current accumulator value and each element.
 */
public inline fun <S> Collection<S>.accumulateRight(operation: (acc: S, T) -> S): S {
    var accumulator: S = this.last() do { while (this往前迭代操作)
accumulator = operation(checkIndexOverflow(index++), accumulator, iterator.next())
} returnaccumulator
/**
 * Accumulates value starting with the first element and applying [operation] from left to right to current accumulator value and each element. Returns null if the collection is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <S, T : S>
IterableViewable<T>.reduceOrNull(operation: (acc: S, T) -> S)?
{ val iterator = this.iterator()
  if (!iterator.hasNext()) return null
  var accumulator = operation(iterator.next(), iterator.next())
  return accumulator
/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. IterableViewable*T.reduceRightOrNull*/
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <S, T : S>
List<T>.reduceRightOrNull(operation: (T, acc: S) -> S)?
{ val iterator = listIterator(size)
  if (!iterator.hasPrevious())
      throw UnsupportedOperationException("Empty list can't be reduced.")
  var accumulator = operation(iterator.previous(), iterator.previous())
  return accumulator
/**
 * Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the list is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
{ val estimatedSize = collectionSizeOrDefault(9)
  if (estimatedSize == 0) return listOf(initial)
  val result = ArrayList<R>(estimatedSize + 1).apply { add(initial) }
  var accumulator = initial
  for (element in this)
      accumulator = operation(accumulator, element)
      result.add(accumulator)
  return result
/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element and current accumulator value that starts with [initial] value.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Iterable<T>.scanReduceIndexed(operation: (index: Int, acc: S, T) -> S): List<S> {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return emptyList()
    var accumulator: S = iterator.next()
    val result = ArrayList<S>(collectionSizeOrDefault(10)).apply { add(accumulator) }
    var index = 1
    while (iterator.hasNext()) {
        accumulator = operation(index++, accumulator, iterator.next())
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right * to each element, its index in the original collection and current accumulator value that starts with the first element of this collection.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * 
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <T> Iterable<T>.chunked(size: Int): List<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this collection into several lists each not exceeding the given [size] and applies the given [transform] function to each.
 * 
 * Note that the list passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * 
 * The last list may have less elements than the given [size].
 * 
 * @param size the number of elements to take in each list, must be positive and can be greater than the number of elements in this collection.
 * 
 * @sample samples.collections.Collections.Transformations.chunked
 */
@SinceKotlin("1.2")
public inline fun <T> Iterable<T>.chunked(size: Int): List<List<T>>
    @Suppress("UNCHECKED_CAST")
    return List(size = size).apply {
        val list = ArrayList<T>(collectionSizeOrDefault(size)).apply { addAll(this@chunked) }
        var index = 0
        while (index < size) {
            var chunk = ArrayList<T>(list.take(size - index)).apply { addAll(this@chunked) }
            add(chunk)
            index += size
        }
    }
Open Source Used In IOX-FogDirector-AC18 1.1.7.0 2178

*\n@SinceKotlin("1.2")\npublic fun <T, R> Iterable<T>.chunked(size: Int, transform: (List<T>) -> R): List<R>
{\n    return windowed(size, size, partialWindows = true, transform = transform)\n}\n
/**
 * Returns a list containing all elements of the original collection without the first occurrence of the given [element].\n * *\npublic operator fun <T> Iterable<T>.minus(element: T): List<T> \n{\n    var removed = false\n    return this.filterTo(result) { \n        if (removed && it == element) { \n            removed = true; false \n        } else true }\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Iterable<T>.minus(elements: Array<out T>): List<T> \n{\n    if (elements.isEmpty()) return this.toList()\n    val other = elements.toHashSet()\n    return this.filterNot { \n        it in other }\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] collection.\n * *\npublic operator fun <T> Iterable<T>.minus(elements: Iterable<T>): List<T> \n{\n    val other = elements.convertToSetForSetOperationWith(this)\n    if (other.isEmpty())
        return this.toList()\n    return this.filterNot { \n        it in other }\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] sequence.\n * *\npublic operator fun <T> Iterable<T>.minus(elements: Sequence<T>): List<T> \n{\n    val other = elements.toHashSet()\n    if (other.isEmpty())
        return this.toList()\n    return this.filterNot { \n        it in other }\n}\n
/**
 * Returns a list containing all elements of the original collection without the first occurrence of the given [element].\n * *\n@kotlin.internal.InlineOnly\npublic inline fun <T> Iterable<T>.minusElement(element: T): List<T> \n{\n    return minus(element)\n}\n
/**
 * Splits the original collection into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, \n * while *second* list contains elements for which [predicate] yielded `false`.\n * *\npublic inline fun <T> Iterable<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> \n{\n    val first = ArrayList<T>()\n    val second = ArrayList<T>()\n    for (element in this) {
        if (predicate(element)) { \n            first.add(element)\n        } else { \n            second.add(element)\n        }\n    }\n    return Pair(first, second)\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Collection<T>.plus(element: T): List<T> \n{\n    val result = ArrayList<T>(size + 1)\n    result.addAll(this)\n    result.add(element)\n    return result\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Collection<T>.plus(elements: Array<out T>): List<T> \n{\n    if (elements is Collection) {
        val result = ArrayList<T>(this.size + elements.size)\n        result.addAll(this)\n        result.addAll(elements)\n        return result\n    } else {
        val result = ArrayList<T>(this.size + elements.size)\n        result.addAll(this)\n        result.addAll(elements)\n        return result\n    }\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Collection<T>.plus(elements: Iterable<T>): List<T> \n{\n    val result = ArrayList<T>(this.size + elements.size)\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Iterable<T>.plus(elements: Elements: Array<out T>): List<T> \n{\n    val result = ArrayList<T>(this.size + elements.size)\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}\n
/**
 * Returns a list containing all elements of the original collection except the elements contained in the given [elements] array.\n * *\npublic operator fun <T> Iterable<T>.plus(elements: Iterable<T>): List<T> \n{\n    val result = ArrayList<T>(this.size + elements.size)\n    result.addAll(this)\n    result.addAll(elements)\n    return result\n}\n
result.addAll(elements)
    return result
}

/**
 * Returns a list containing all elements of the original collection and then the given [elements] sequence.
 */
public operator fun <T> Iterable<T>.plus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>()
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a list containing all elements of the original collection and then the given [elements] sequence.
 */
public operator fun <T> Collection<T>.plus(elements: Sequence<T>): List<T> {
    val result = ArrayList<T>(this.size + 10)
    result.addAll(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a list containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Iterable<T>.plusElement(element: T): List<T> {
    return plus(element)
}

/**
 * Returns a list containing all elements of the original collection and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Collection<T>.plusElement(element: T): List<T> {
    return plus(element)
}

/**
 * Returns a list of snapshots of the window of the given [size] sliding along this collection with the given [step], where each snapshot is a list.
 * Several last lists may have less elements than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved
 * @param sample collections.Sequences.Transformations.takeWindows
 * @param SinceKotlin("1.2")
 */
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<List<T>>(resultCapacity)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (!partialWindows && windowSize < size) break
            result.add(List(windowSize) { this[it + index] })
            index += step
        }
        return result
    }
    val result = ArrayList<List<T>>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = false).forEach {
        result.add(it)
    }
    return result
}

/**
 * Returns a list of results of applying the given [transform] function to an each list representing a view over the window of the given [size] sliding along this collection with the given [step]. Several last lists may have less elements than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this collection.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any, by default `false` which means partial windows won't be preserved
 * @param sample collections.Sequences.Transformations.averageWindows
 * @param SinceKotlin("1.2")
 */
    checkWindowSizeStep(size, step)
    if (this is RandomAccess && this is List) {
        val thisSize = this.size
        val resultCapacity = thisSize / step + if (thisSize % step == 0) 0 else 1
        val result = ArrayList<R>(resultCapacity)
        val window = MovingSubList(this)
        var index = 0
        while (index in 0 until thisSize) {
            val windowSize = size.coerceAtMost(thisSize - index)
            if (!partialWindows && windowSize < size) break
            window.move(index, index + windowSize)
            result.add(transform(window))
            index += step
        }
        return result
    }
    val result = ArrayList<R>()
    windowedIterator(iterator(), size, step, partialWindows, reuseBuffer = true).forEach {
        result.add(transform(it))
    }
    return result
}

/**
 * Returns a list of pairs built from the elements of `this` collection and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * @param other the other array to pair with
 */
public infix fun <T, R> Iterable<T>.zip(other: Array<out R>): List<Pair<T, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}
[transform] function applied to each pair of elements.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.

The returned list has length of the shortest collection.
public fun Iterable<Byte>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Iterable<Short>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Iterable<Int>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Iterable<Long>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Iterable<Float>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns an average value of elements in the collection.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Iterable<Double>.average(): Double {
    var sum = Double(0.0)
    var count = Int(0)
    for (element in this) {
        sum += element
        if (count > Int.MAX_VALUE) return Double.NaN
        count = Int(count + 1)
    }
    return sum / count
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Iterable<Byte>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Iterable<Short>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Iterable<Int>.sum(): Int {
    var sum = Int(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Iterable<Long>.sum(): Long {
    var sum = Long(0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Iterable<Float>.sum(): Float {
    var sum = Float(0.0)
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the collection.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Iterable<Double>.sum(): Double {
    var sum = Double(0.0)
    for (element in this) {
        sum += element
    }
    return sum
}

// Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
// Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```java
val iterator = collection.iterator()
var index = 0
while (iterator.hasNext()) {
    array[index++] = iterator.next().unsafeCast<T>()
}
if (index < array.size) {
    array[index] = null.unsafeCast<T>()
}
return array
}

/**
 * Returns an immutable list containing only the specified object [element].
 */
val listOf(element: T): List<T> = arrayListOf(element)

/**
 * Returns an immutable set containing only the specified object [element].
 */
val setOf(element: T): Set<T> = hashSetOf(element)

/**
 * Returns an immutable map, mapping only the specified key to the
 * specified value.
 */
val mapOf(pair: Pair<K, V>): Map<K, V> = hashMapOf(pair)

/**
 * Fills the list with the provided [value].
 */
@SinceKotlin("1.2")
val MutableList<T>.fill(value: T): Unit {
    for (index in 0..lastIndex) {
        this[index] = value
    }
}

/**
 * Randomly shuffles elements in this list.
 */
@SinceKotlin("1.2")
val MutableList<T>.shuffle(): Unit = shuffle(Random)

/**
 * Returns a new list with the elements of this list randomly shuffled.
 */
@SinceKotlin("1.3")
@PublishedApi
@JsName("arrayCopy")
internal fun <T> arrayCopy(source: Array<out T>, destination: Array<in T>, destinationOffset: Int, startIndex: Int, endIndex: Int) {
    AbstractList.checkRangeIndexes(startIndex, endIndex, source.size)
    AbstractList.checkRangeIndexes(destinationOffset, destinationOffset + rangeSize, destination.size)

    if (js("ArrayBuffer").isView(source) && js("ArrayBuffer").isView(destination)) {
        val subrange = source.asDynamic().subarray(startIndex, endIndex)
        destination.asDynamic().set(subrange, destinationOffset)
    } else {
        if (source !== destination || destinationOffset <= startIndex) {
            for (index in 0 until rangeSize) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        } else {
            for (index in rangeSize - 1 downTo 0) {
                destination[destinationOffset + index] = source[startIndex + index]
            }
        }
    }
}
```

---

**Open Source Used In IOx-FogDirector-AC18 1.17.0 2182**

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

```kotlin
package kotlin.collections

/**
 * Returns the given iterator itself. This allows to use an instance of iterator in a `for` loop.
 * @sample samples.collections.Iterators.iterator
 */
@kotlin.internal.InlineOnly
public inline operator fun <T> Iterator<T>.iterator(): Iterator<T> = this

/**
 * Returns an [Iterator] that wraps each element produced by the original iterator into an
 * [IndexedValue] containing the index of that element and the element itself.
 * @sample samples.collections.Iterators.withIndex
 */
public fun <T> Iterator<T>.withIndex(): Iterator<IndexedValue<T>> = IndexingIterator(this)

/**
 * Performs the given operation on each element of this [Iterator].
 * @sample samples.collections.Iterators.forEach
 */
public inline fun <T> Iterator<T>.forEach(operation: (T) -> Unit): Unit {
    for (element in this) operation(element)
}

/**
 * `IndexingIterator` transforming original `iterator` into iterator of
 * [IndexedValue], counting index from zero.
 */
internal class IndexingIterator<out T>(private val iterator: Iterator<T>) : Iterator<IndexedValue<T>> {
    private var index = 0
    final override fun hasNext(): Boolean = iterator.hasNext()
    final override fun next(): IndexedValue<T> = IndexedValue(checkIndexOverflow(index++), iterator.next())
}
```

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

```kotlin
package kotlin.comparisons

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See:
// https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*

/**
 * Returns the greater of two values.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public expect fun <T : Comparable<T>> maxOf(a: T, b: T): T

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Byte, b: Byte): Byte

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Short, b: Short): Short

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Int, b: Int): Int

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Long, b: Long): Long

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Float, b: Float): Float

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun maxOf(a: Double, b: Double): Double

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, c: T, comparator: Comparator<in T>): T {
    return maxOf(a, maxOf(b, c, comparator), comparator)
}

/**
 * Returns the greater of two values
 * according to the order specified by the given 
 * [comparator].
 */
@SinceKotlin("1.1")
public fun <T> maxOf(a: T, b: T, comparator: Comparator<in T>): T {
    return maxOf(a, maxOf(b, comparator), comparator)
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2183
returns the first one.

```kotlin
@SinceKotlin("1.1")
public expect fun <T : Comparable<T>> minOf(a: T, b: T): T

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Byte, b: Byte): Byte

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Short, b: Short): Short

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Int, b: Int): Int

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Long, b: Long): Long

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Float, b: Float): Float

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
public expect fun <T: Comparable<T>> minOf(a: T, b: T, c: T): T

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Byte, b: Byte, c: Byte): Byte

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Short, b: Short, c: Short): Short

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Int, b: Int, c: Int): Int

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Long, b: Long, c: Long): Long

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public expect inline fun minOf(a: Float, b: Float, c: Float): Float

/**
 * Returns the smaller of three values according to the order specified by the given
 * comparator.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: T, b: T, c: T, comparator: Comparator<in T>): T {
    return minOf(a, minOf(b, c, comparator), comparator)
}

/**
 * Returns the smaller of two values according to the order specified by the given
 * comparator.
 * If values are equal, returns the first one.
 */
@SinceKotlin("1.1")
public expect fun minOf(a: T, b: T, comparator: Comparator<in T>): T {
    return if (comparator.compare(a, b) <= 0) a else b
}
```

```
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("MapsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.
import kotlin.ranges.contains
import kotlin.ranges.reversed
import kotlin.ranges.reversed

public package
kotlin.collections

//
// Use of this source code is governed by the Apache 2.0 license that can be
// found in the license/LICENSE.txt file.

import @file:kotlin.jvm.JvmMultifileClass
import @file:kotlin.jvm.JvmName("MapsKt")

unpackage
kotlin.collections

//
// This is a generated file.
//
// DO NOT EDIT IT MANUALLY.
```
samples.collections.Maps.Transformations.mapToList

*/
public inline fun <K, V, R> Map<out K, V>.map(transform: (Map.Entry<K, V>) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing only the non-null results of applying the given [transform] function
 * to each entry in the original map.
 */
public inline fun <K, V, R> Map<out K, V>.mapNotNull(transform: (Map.Entry<K, V>) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(), transform)
}

/**
 * Applies the given [transform] function to each entry in the original map and appends only
 * the non-null results to the given [destination].
 */
public inline fun <K, V, R> Map<out K, V>.mapTo(destination: C<in R>, transform: (Map.Entry<K, V>) -> R?): C<in R> {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns `true` if all entries match the given [predicate].
 */
public inline fun <K, V> Map<out K, V>.all(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return true
    for (element in this)
        if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if map has at least one entry.
 */
public fun <K, V> Map<out K, V>.any(): Boolean {
    return !isEmpty()
}

/**
 * Returns `true` if at least one entry matches the given [predicate].
 */
public inline fun <K, V> Map<out K, V>.any(predicate: (Map.Entry<K, V>) -> Boolean): Boolean {
    if (isEmpty()) return false
    for (element in this)
        if (predicate(element)) return true
    return false
}

/**
 * Performs the given [action] on each entry and returns the map itself afterwards.
 */
@SinceKotlin("1.1")
public inline fun <K, V, M : Map<out K, V>> M.onEach(action: (Map.Entry<K, V>) -> Unit): M {
    return apply { for (element in this)
        action(element) }
}

/**
 * Creates an [Iterable] instance that wraps the original map returning its entries when being iterated.
 */
@kotlin.internal.InlineOnly
public inline fun <K, V> Map<out K, V>.asIterable(): Iterable<Map.Entry<K, V>> {
    return entries
}

/**
 * Creates a [Sequence] instance that wraps the original map returning its entries when being iterated. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SequencesKt")

package kotlin.sequences

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib/kotlin
//

import kotlin.random.*

/**
 * Returns `true` if [element] is found in the sequence. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element)
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this sequence. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element)
}

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this sequence. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element)
}

/**
 * Returns first element.
 * @throws [NoSuchElementException] if the sequence is empty. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element)
}

/**
 * Returns first element matching the given [predicate].
 * @throws [NoSuchElementException] if no such element is found. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.findFirst(predicate: (T) -> Boolean): T {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Sequence contains no element matching the predicate.")
}

/**
 * Returns index of [element], or -1 if the sequence does not contain element. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.indexOf(element: T): Int {
    var count = 0
    for (item in this) {
        if (element == item) return count
        count++
    }
    return -1
}

/**
 * Returns the first element matching the given [predicate], or `null` if the sequence does not contain element. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.indexOf(element: T): Int {
    var count = 0
    for (item in this) {
        if (element == item) return count
        count++
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the sequence does not contain element. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.indexOf(element: T): Int {
    var count = 0
    for (item in this) {
        if (element == item) return count
        count++
    }
    return -1
}

/**
 * Returns first index of [element], or -1 if the sequence does not contain element. * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
 * The operation is _terminal_ and has no operator for run <@kotlin.internal.OnlyInputTypes T> Sequence<T>.contains(element: T): Boolean {
    return indexOf(element) >= 0
}

@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.indexOf(element: T): Int {
    var count = 0
    for (item in this) {
        if (element == item) return count
        count++
    }
    return -1
}
public inline fun <T> Sequence<T>.indexOfFirst(predicate: (T) -> Boolean): Int {
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            return index
        index++
    }
    return -1
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the sequence does not
 * contain such element.
 */
public inline fun <T> Sequence<T>.indexOfLast(predicate: (T) -> Boolean): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (predicate(item))
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the sequence is empty.
 */
public fun <T> Sequence<T>.last(): T {
    val iterator = iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Sequence is empty.")
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}

/**
 * Returns the last element matching the given [predicate], or `null` if no such element is found.
 */
public inline fun <T> Sequence<T>.lastOrNull(predicate: (T) -> Boolean): T? {
    var last: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            last = element
            found = true
        }
    }
    return if (found) last
    else null
}

/**
 * Returns last index of [element], or -1 if the sequence does not contain element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.lastIndexOf(element: T): Int {
    var lastIndex = -1
    var index = 0
    for (item in this) {
        checkIndexOverflow(index)
        if (element == item)
            lastIndex = index
        index++
    }
    return lastIndex
}

/**
 * Returns the last element, or `null` if the sequence is empty.
 */
public fun <T> Sequence<T>.lastOrNull(): T? {
    val iterator = iterator()
    if (!iterator.hasNext())
        return null
    var last = iterator.next()
    while (iterator.hasNext())
        last = iterator.next()
    return last
}

/**
 * Returns the single element matching the given [predicate], or `null` if element was not
 * found or more than one element was found.
 */
public inline fun <T> Sequence<T>.singleOrNull(predicate: (T) -> Boolean): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns single element, or throws an exception if the sequence contains more than one element.
 */
public fun <T> Sequence<T>.single(): T {
    val iterator = iterator()
    if (!iterator.hasNext())
        throw NoSuchElementException("Sequence contains more than one matching element.")
    val single = iterator.next()
    if (iterator.hasNext())
        throw IllegalArgumentException("Sequence contains more than one matching element.")
    @Suppress("UNCHECKED_CAST")
    return single as T
}

/**
 * Returns single element, or throws an exception if the sequence contains more than one element.
 */
public fun <@kotlin.internal.OnlyInputTypes T> Sequence<T>.singleOrNull(): T? {
    var single: T? = null
    var found = false
    for (element in this) {
        if (found) return null
        single = element
        found = true
    }
    if (!found) return null
    return single
}
Sequence<T> { n require(n >= 0) { "Requested element count $n is less than zero." } return when { n == 0 -> this
    this is DropTakeSequence -> this.drop(n)
    else -> DropSequence(this, n)
} n

Returns a sequence containing all elements except first elements that satisfy the given [predicate]. The operation is _intermediate_ and _stateless_.

samples.collections.Collections.Transformations.drop(n) *@public fun T Sequence<T>.dropWhile(predicate: (T) -> Boolean): Sequence<T> { n return DropWhileSequence(this, predicate)
**

/**
 * Returns a sequence containing all elements not matching the given [predicate].
 * @throws IllegalArgumentException if [n] is negative.
 * @param n * @sample samples.collections.Collections.Transformations.take(n) */

*\public fun <T> Sequence<T>.takeWhile(predicate: (T) -> Boolean): Sequence<T> {*
    return TakeWhileSequence(this, predicate)\n} \n/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * The operation is _intermediate_ and _stateful_.\n */
\npublic fun <T> Sequence<T>.sorted(): Sequence<T> {\n    return sortedListIterator()\n} \n/**
 * Returns a sequence that yields elements of this sequence sorted according to natural sort order of the value returned by specified [selector] function.\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * The operation is _intermediate_ and _stateful_.\n */
\npublic fun <T : Comparable<T>> Sequence<T>.sortedDescending(): Sequence<T> {\n    return sortedListIterator()\n} \n/**
 * Returns a sequence that yields elements of this sequence sorted according to their natural sort order.\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * The operation is _intermediate_ and _stateful_.\n */
\npublic fun <T> Sequence<T>.sortedBy(crossinline selector: (T) -> R?): Sequence<T> {\n    return sortedListIterator()\n} \n/**
 * Returns a sequence that yields elements of this sequence sorted according to natural sort order of the value returned by specified [selector] function.\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * The operation is _intermediate_ and _stateful_.\n */
\npublic fun <T : Comparable<T>> Sequence<T>.sortedByDescending(crossinline selector: (T) -> R?): Sequence<T> {\n    return sortedListIterator()\n} \n/**
 * Returns a sequence that yields elements of this sequence sorted according to the specified [comparator].\n * The sort is _stable_. It means that equal elements preserve their order relative to each other after sorting.\n * The operation is _intermediate_ and _stateful_.\n */
\npublic fun <T> Sequence<T>.sortedWith(comparator: Comparator<in T>): Sequence<T> {\n    return sortedListIterator()\n} \n/**
 * Returns a [Map] containing key-value pairs provided by [transform] function applied to elements of the given sequence.\n * The operation is _terminal_.\n */
\npublic inline fun <T, K, V> Sequence<T>.associate(transform: (T) -> Pair<K, V>): Map<K, V> {\n    return associateTo(LinkedHashMap<K, V>(), transform)\n} \n/**
 * Returns a [Map] containing the elements from the given sequence indexed by the key returned from [keySelector] function applied to each element.\n * The operation is _terminal_.\n */
\npublic inline fun <T, K> Sequence<T>.associateBy(keySelector: (T) -> K): Map<K, T> {\n    return associateByTo(LinkedHashMap<K, T>(), keySelector)\n} \n/**
 * Returns a [Map] containing the values provided by [valueTransform] and indexed by [keySelector] functions applied to elements of the given sequence.\n * The operation is _terminal_.\n */
\npublic inline fun <T, K, M :
MutableMap<in K, in T>Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this) {
        destination.put(keySelector(element), element)
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable map with key-value pairs,
 * where key is provided by the [keySelector] function and
 * value is provided by the [valueTransform] function applied to elements of the
given sequence.
 * If any two elements would have the same key returned by [keySelector] the last one gets
added to the map.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.associateByToWithValueTransform
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateByTo(destination: M, keySelector: (T) -> K,
valueTransform: (T) -> V): M {
    for (element in this) {
        destination.put(keySelector(element), valueTransform(element))
    }
    return destination
}

/**
 * Populates and returns the [destination] mutable
 * map with key-value pairs
 * provided by [transform] function applied to each element of the given sequence.
 * If any two elements are equal, the last one overwrites the former value in the
 * map.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.associateTo
 */
public inline fun <T, K, V, M : MutableMap<in K, in V>> Sequence<T>.associateTo(destination: M, transform: (T) -> Pair<K, V>): M {
    for (element in this) {
        destination += transform(element)
    }
    return destination
}

/**
 * Returns a [Map] where keys are elements from the given sequence and values are
 * produced by the [valueSelector] function applied to each element.
 * If any two elements are equal, the last one gets added to the map.
 * The returned map preserves the entry iteration order of the original sequence.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.associateWith
 */
@SinceKotlin("1.3")
public inline fun <K, V> Sequence<K>.associateWith(valueSelector: (K) -> V): Map<K, V> {
    val result = LinkedHashMap<K, V>()
    return associateWithTo(result,
valueSelector)
}

/**
 * Returns a [MutableMap] filled with all elements of this
 * sequence.
 * The returned map preserves the element iteration order of the original sequence.
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.associateWithTo
 */
@SinceKotlin("1.3")
public inline fun <K, V, M : MutableMap<in K, in V>> Sequence<K>.associateWithTo(destination: M, valueSelector: (K) -> V): M {
    for (element in this) {
        destination.put(element, valueSelector(element))
    }
    return destination
}

/**
 * Appends all elements to the given [destination] collection.
 * The operation is _terminal_.
 */
public fun <T, C : MutableCollection<in T>> Sequence<T>.toCollection(destination: C): C {
    for (item in this) {
        destination.add(item)
    }
    return destination
}

/**
 * Returns a [HashSet] of all elements.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toHashSet(): HashSet<T> {
    return toCollection(HashSet<T>())
}

/**
 * Returns a [List] containing all elements.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toList(): List<T> {
    return this.toMutableList().optimizeReadOnlyList()
}

/**
 * Returns a single sequence of all elements
 * from results of [transform] function being invoked on each element of original sequence.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Collections.Transformations.flatMap
 */
public fun <T, R> Sequence<T>.flatMap(transform: (T) -> Sequence<R>): Sequence<R> {
    return FlatteningSequence(this, transform, { it.iterator() })
}

/**
 * Appends all elements yielded from results of
 * [transform] function being invoked on each element of original sequence, to the given [destination].
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.flatMapTo
 */
public inline fun <T, R, C : MutableCollection<in R>> Sequence<T>.flatMapTo(destination: C, transform: (T) -> Sequence<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}
the original sequence by the key returned by the given [keySelector] function\n * applied to each element and returns a map where each group key is associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original sequence.\n
 The operation is _terminal_\n * \n * @sample samples.collections.Collections.Transformations.groupBy\n */
public inline fun <T, K> Sequence<T>.groupBy(keySelector: (T) -> K): Map<K, List<T>> {
    return groupByTo(LinkedHashMap<K, MutableList<T>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original sequence by the key returned by the given [keySelector] function\n * @return The \[destination\] map each group key associated with a list of corresponding elements.\n * The operation is _terminal_\n * @sample samples.collections.Collections.Transformations.groupBy\n */
public inline fun <T, K, M : MutableMap<in K, MutableList<T>>> Sequence<T>.groupByTo(destination: M, keySelector: (T) -> K): M {
    for (element in this)
    {
        val key = keySelector(element)
        val list = destination.getOrPut(key) { ArrayList<T>() }
        list.add(element)
    }
    return destination
}

/**
 * Creates a [Grouping] source from a sequence to be used later with one of group-and-fold operations\n * using the specified [keySelector] function to extract a key from each element.\n * The operation is _terminal_\n * @sample samples.collections.Collections.Transformations.groupBy\n */
@SinceKotlin("1.1")
public inline fun <T, K> Sequence<T>.groupingBy(crossinline keySelector: (T) -> K): Grouping<T, K> {
    return object : Grouping<T, K> {
        override fun sourceIterator(): Iterator<T> = this@groupingBy.iterator()
        override fun keyOf(element: T): K = keySelector(element)
    }
}

/**
 * Returns a sequence containing the results of applying the given [transform] function\n * to each element and its index in the original sequence.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n * The operation is _intermediate_ and _stateless_ \n * @sample samples.collections.Collections.Transformations.map\n */
public fun <T, R> Sequence<T>.map(transform: (T) -> R): Sequence<R> {
    return TransformingSequence(this, transform)
}

/**
 * Applies the given [transform] function\n * to each element and its index in the original sequence.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n * The operation is _intermediate_ and _stateless_ \n * @sample samples.collections.Collections.Transformations.map\n */
public fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}

/**
 * Applies the given [transform] function\n * to each element and its index in the original sequence.\n * @sample samples.collections.Collections.Transformations.map\n */
public fun <T, R : Any> Sequence<T>.mapIndexed(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}

/**
 * Applies the given [transform] function\n * to each element and its index in the original sequence.\n * @sample samples.collections.Collections.Transformations.map\n */
public fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}

/**
 * Applies the given [transform] function\n * to each element and its index in the original sequence.\n * @sample samples.collections.Collections.Transformations.map\n */
public fun <T, R : Any> Sequence<T>.mapIndexedNotNull(transform: (index: Int, T) -> R?): Sequence<R> {
    return TransformingIndexedSequence(this, transform).filterNotNull()
}
[destination].

* @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.

* The operation is _terminal_.

*/
public inline fun

<

<T, R : Any, C : MutableCollection<R>> Sequence<T>.mapIndexedNotNullTo(destination: C, transform: (index: Int, T) -> R?): C {
    forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }
    return destination
}

/**
 * Applies the given [transform] function to each element and its index in the original sequence and appends the results to the given [destination].
 * @param [transform] function that takes the index of an element and the element itself and returns the result of the transform applied to the element.
 * The operation is _terminal_.
 */
public inline fun <T, R : Any, C : MutableCollection<R>> Sequence<T>.mapIndexedTo(destination: C, transform: (index: Int, T) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(checkIndexOverflow(index++), item))
    return destination
}

/**
 * Returns a sequence containing only the non-null results of applying the given [transform] function to each element in the original sequence.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T, R : Any> Sequence<T>.mapNotNull(transform: (T) -> R?): Sequence<R> {
    return TransformingSequence(this, transform).filterNotNull()
}

/**
 * Applies the given [transform] function to each element in the original sequence and appends only the non-null results to the given [destination].
 * The operation is _terminal_.
 */
public inline fun <T, R : Any, C : MutableCollection<R>> Sequence<T>.mapNotNullTo(destination: C, transform: (T) -> R?): C {
    forEach { element -> transform(element)?.let { destination.add(it) } }
    return destination
}

/**
 * Applies the given [transform] function to each element of the original sequence and appends the results to the given [destination].
 * The operation is _terminal_.
 */
public inline fun <T, R, C : MutableCollection<R>> Sequence<T>.mapTo(destination: C, transform: (T) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a sequence that wraps each element of the original sequence into an [IndexedValue] containing the index and the element itself.
 * The operation is _intermediate_ and _stateless_.
 */
public fun <T> Sequence<T>.withIndex(): Sequence<IndexedValue<T>> {
    return IndexingSequence(this)
}

/**
 * Returns a mutable set containing all distinct elements from the given sequence.
 * The returned set preserves the element iteration order of the original sequence.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.toMutableSet(): MutableSet<T> {
    val set = LinkedHashSet<T>()
    for (item in this)
        set.add(item)
    return set
}

/**
 * Returns `true` if all elements match the given [predicate].
 * The operation is _terminal_.
 * @sample samples.collections.Aggregates.all
 */
public inline fun <T> Sequence<T>.all(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (!predicate(element)) return false
    return true
}

/**
 * Returns `true` if sequence has at least one element.
 * The operation is _terminal_.
 * @sample samples.collections.Aggregates.any
 */
public fun <T> Sequence<T>.any(): Boolean {
    return iterator().hasNext()
}

/**
 * Returns `true` if at least one element matches the given [predicate].
 * The operation is _terminal_.
 * @sample samples.collections.Aggregates.anyWithPredicate
 */
public inline fun <T> Sequence<T>.any(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements matching the given [predicate].
 * The operation is _terminal_.
 * @sample samples.collections.Aggregates.count
 */
public fun <T> Sequence<T>.count(predicate: (T) -> Boolean): Int {
    var count = 0
    for (element in this)
        if (predicate(element)) count++
    return count
}

/**
 * Returns the number of elements in this sequence.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.count(): Int {
    var count = 0
    for (element in this)
        count++
    return count
}
Sequence<Float>.min(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}

Sequence<Double>.min(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var min = iterator.next()
    if (min.isNaN()) return min
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (e.isNaN()) return e
        if (min > e) min = e
    }
    return min
}

Sequence<T>.max(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

Sequence<Float>.max(): Float? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

Sequence<Double>.max(): Double? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var max = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (max < e) max = e
    }
    return max
}

Sequence<T>.maxWith(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var maxElem = iterator.next()
    while (iterator.hasNext()) {
        val e = iterator.next()
        if (comparator.compare(maxElem, e) < 0) maxElem = e
    }
    return maxElem
}

Sequence<T>.maxBy(selector: (T) -> R): R {
    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

Sequence<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var accumulator = initial
    for (index in 0 until count) accumulator = operation(checkIndexOverflow(index), accumulator, element)
    return accumulator
}

Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

Sequence<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

Sequence<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(checkIndexOverflow(index), accumulator, element)
    return accumulator
}

Sequence<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

Sequence<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (item in this) action(checkIndexOverflow(index++), item)
}

Sequence<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

Array<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (element in this) action(checkIndexOverflow(index++), element)
}

Array<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}

Array<T>.foldIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(checkIndexOverflow(index), accumulator, element)
    return accumulator
}

Array<T>.fold(initial: R, operation: (acc: R, T) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

Array<T>.forEachIndexed(action: (index: Int, T) -> Unit): Unit {
    var index = 0
    for (element in this) action(checkIndexOverflow(index++), element)
}

Array<T>.forEach(action: (T) -> Unit): Unit {
    for (element in this) action(element)
}
iterator.next() \n    if (min.isNaN()) return min \n    while (iterator.hasNext()) \n        val e = iterator.next() \n        if (e.isNaN()) return e \n        if (min > e) min = e \n    \n    return min
}

/**
 * Returns the smallest element or null if there are no elements.
 * The operation is _terminal_.
 */
public fun <T : Comparable<T>> Sequence<T>.min(): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var minElem = iterator.next()
    if (!iterator.hasNext()) return minElem
    var minValue = minElem
    do {
        val e = iterator.next()
        val v = e
        if (minValue > v) {
            minValue = v
            minElem = e
        }
    } while (iterator.hasNext())
    return minElem
}

/**
 * Returns the first element having the smallest value according to the provided [comparator] or null if there are no elements.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.minWith(comparator: Comparator<in T>): T? {
    val iterator = iterator()
    if (!iterator.hasNext()) return null
    var accumulator: T = iterator.next()
    while (iterator.hasNext()) {
        accumulator = comparator.compare(accumulator, iterator.next()) \n        \n    }
    return accumulator
}

/**
 * Returns `true` if the sequence has no elements.
 * The operation is _terminal_.
 */
public fun <T> Sequence<T>.none(): Boolean {
    return !iterator().hasNext()
}

/**
 * Returns `true` if no elements match the given [predicate].
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
public inline fun <T> Sequence<T>.none(predicate: (T) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to
 * current accumulator value and each element. Returns null if the sequence is empty.
 * The operation is _terminal_.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) return null
    var accumulator: S = iterator.next()
    while (iterator.hasNext()) {
        accumulator = operation(accumulator, iterator.next()) \n        \n    }
    return accumulator
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to
 * current accumulator value and each element. Returns null if the sequence is empty.
 * The operation is _terminal_.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.") \n    var accumulator: S = iterator.next() \n    while (iterator.hasNext()) \n        accumulator = operation(accumulator, iterator.next()) \n        \n    return accumulator
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to
 * current accumulator value and each element. Returns null if the sequence is empty.
 * The operation is _terminal_.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun <S, T : S> Sequence<T>.reduceOrNull(operation: (acc: S, T) -> S): S? {
    val iterator = this.iterator()
    if (!iterator.hasNext()) throw UnsupportedOperationException("Empty sequence can't be reduced.") \n    var accumulator: S = iterator.next() \n    while (iterator.hasNext()) \n        accumulator = operation(accumulator, iterator.next()) \n        \n    return accumulator
}
function should not be mutated; otherwise it would affect the previous value in resulting sequence. The initial value should also be immutable (or should not be mutated); otherwise it would affect the previous value in resulting sequence. The initial value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence’s lazy nature.

- @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
- The operation is _intermediate_ and _stateless_.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T, R> Sequence<T>.scan(initial: R, operation: (acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var accumulator = initial
        for (element in this@scan) {
            accumulator = operation(accumulator, element)
            yield(accumulator)
        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence. The initial value should also be immutable (or should not be mutated) as it may be passed to [operation] function later because of sequence’s lazy nature.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <T, R> Sequence<T>.scanIndexed(initial: R, operation: (index: Int, acc: R, T) -> R): Sequence<R> {
    return sequence {
        yield(initial)
        var index = 0
        var accumulator = initial
        for (element in this@scanIndexed) {
            accumulator = operation(checkIndexOverflow(index++), accumulator, element)
            yield(accumulator)
        }
    }
}

/**
 * Returns a sequence containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original sequence and current accumulator value that starts with the first element of this sequence.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting sequence.
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun <S, T : S> Sequence<T>.scanReduce(operation: (acc: S, T) -> S): Sequence<S> {
    return sequence {
        val iterator = iterator()
        if (iterator.hasNext()) {
            var accumulator: S = iterator.next()
            yield(accumulator)
            while (iterator.hasNext()) {
                accumulator = operation(iterator.next(), accumulator)
                yield(accumulator)
            }
        }
    }
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumBy(selector: (T) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each element in the sequence.
 * The operation is _terminal_.
 */
public inline fun <T> Sequence<T>.sumByDouble(selector: (T) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}
```
Returns an original collection containing all the non-`null` elements, throwing an [IllegalArgumentException] if there are any `null` elements. The operation is _intermediate_ and _stateless_.

```kotlin
return sum
}

/**
 * Returns an original collection containing all the non-`null` elements, throwing an
 * `[IllegalArgumentException]` if there are any `null` elements. The last list in the resulting sequence may have less
 * elements than the given [size].
 * @param size The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param sample size The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param size The operation is _intermediate_ and _stateful_.
 */
public fun <T : Any> Sequence<T?>.requireNoNulls(): Sequence<T> {
    return map { it ?: throw IllegalArgumentException("null element found in $this.") }
}

/**
 * Splits this sequence into a sequence of lists each not exceeding the given [size].
The last list in the resulting sequence may have less elements than the given [size].
 * @param size The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param size The operation is _intermediate_ and _stateful_.
 */
@SinceKotlin("1.2")
public fun <T> Sequence<T>.chunked(size: Int): Sequence<List<T>> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this sequence into several lists each not exceeding
 * the given [size] and applies the given [transform] function to an each.
 * @param size The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param size The operation is _intermediate_ and _stateful_.
 */
@SinceKotlin("1.2")
public fun <T, R> Sequence<T>.chunked(size: Int, transform: (List<T>) -> R): Sequence<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given
 * [element].
 * @param size The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param size The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(element: T): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            var removed = false
            return this@minus.filter { if (!removed && it == element) {
                removed = true; false } else true }.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given 
 * [elements] array.
 * @param elements The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param elements The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(elements: Array<out T>): Sequence<T> {
    if (elements.isEmpty()) return this
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.toHashSet()
            if (other.isEmpty())
                return this@minus.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given 
 * [elements] collection.
 * @param elements The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param elements The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(elements: Iterable<T>): Sequence<T> {
    return object: Sequence<T> {
        override fun iterator(): Iterator<T> {
            val other = elements.convertToSetForSetOperation()
            if (other.isEmpty())
                return this@minus.iterator()
        }
    }
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given 
 * [elements] sequence.
 * @param elements The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param elements The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    val var removed = false
    return this@minus.filter { if (!removed & it == element) {
        val removed = true; false } else true }.iterator()
}

/**
 * Returns a sequence containing all elements of original sequence except the elements contained in the given 
 * [elements] sequence.
 * @param elements The number of elements to take in each list, must be positive and
can be greater than the number of elements in this sequence.
 * @param elements The operation is _intermediate_ and _stateful_.
 */
public operator fun <T> Sequence<T>.minus(elements: Sequence<T>): Sequence<T> {
    val var removed = false
    return this@minus.filter { if (!removed & it == element) {
        val removed = true; false } else true }.iterator()
return object: Sequence<T> { override fun iterator(): Iterator<T> {
        val other = elements.toHashSet()
        if (other.isEmpty())
            return this@minus.iterator()
        else
            return this@minus.filterNot { it in other }.iterator()
    }
}

/**
 * Returns a sequence containing all elements of the original sequence without the first occurrence of the given [element].
 * The operation is _intermediate_ and _stateless_.
 */
@kotlin.internal.InlineOnly
public inline fun <T, element: T> Sequence<T>.minusElement(element: T): Sequence<T> {
    return return minus(element)
}

/**
 * Splits the original sequence into pair of lists, where *first* list contains elements for which [predicate] yielded `true`, and *second* list contains elements for which [predicate] yielded `false`. The operation is _terminal_.
 */
@SinceKotlin("1.2")
    return windowedSequence(size, step, partialWindows, reuseBuffer = false)
}

/**
 * Returns a sequence of snapshots of the window of the given [size] sliding along this sequence with the given [step], where each window is a list. Several last lists may have less elements than the given [size]. Both [size] and [step] must be positive and can be greater than the number of elements in this sequence.
 */
@SinceKotlin("1.2")
public inline fun <T, element: T> Sequence<T>.plusElement(element: T): Sequence<T> {
    return plus(element)
}

/**
 * Returns a sequence of results of applying the given [transform] function to an each list representing a view over the window of the given [size] sliding along this sequence with the given [step].
 */
@SinceKotlin("1.2")
    return windowed(size, Int, partialWindows, Boolean = false)
}

/**
 * Returns a sequence containing all elements of the original sequence and then the given [element].
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusElement(element: T): Sequence<T> {
    return plusElement(element)
}

/**
 * Returns a sequence containing all elements of the original sequence and then the given [elements] array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusArray(elements: Array<T>): Sequence<T> {
    return plusArray(elements)
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusCollection(elements: Iterable<T>): Sequence<T> {
    return plusCollection(elements)
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] sequence.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusSequence(elements: Sequence<T>): Sequence<T> {
    return plusSequence(elements)
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] array, while *second* list contains elements for which [predicate] yielded `false`. The operation is _terminal_.
 */
@SinceKotlin("1.2")
public inline fun <T, element: T> Sequence<T>.partition(predicate: (T) -> Boolean): Pair<List<T>, List<T>> {
    val first = ArrayList<T>()
    val second = ArrayList<T>()
    for (element in this) {
        if (predicate(element)) {
            first.add(element)
        } else {
            second.add(element)
        }
    }
    return Pair(first, second)
}

/**
 * Returns a sequence containing all elements of the original sequence and then the given [element].
 */
public operator fun <T> Sequence<T>.plusArray(elements: Array<T>): Sequence<T> {
    return sequenceOf(this, sequenceOf(elements.asList())).flatten()
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] array.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusArray(elements: Array<T>): Sequence<T> {
    return plusArray(elements)
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] collection.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusCollection(elements: Iterable<T>): Sequence<T> {
    return sequenceOf(this, elements.asSequence()).flatten()
}

/**
 * Returns a sequence containing all elements of the original sequence and then all elements of the given [elements] sequence.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Sequence<T>.plusSequence(elements: Sequence<T>): Sequence<T> {
    return sequenceOf(this, sequenceOf(elements)).flatten()
}
partialWindows controls whether or not to keep partial windows in the end if any. By default "false" which means partial windows won't be preserved

}

/** * Returns a sequence of values built from the elements of `this` sequence and the [other] sequence with the same index.
 * The resulting sequence ends as soon as the shortest input sequence ends.
 *
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Sequences.Transformations.averageWindows
 */
public fun <T, R> Sequence<T>.zip(other: Sequence<T>, transform: (a: T, b: R) -> R): Sequence<R> { return MergingSequence(this, other, transform)
}

/** * Returns a sequence of values built from the elements of `this` sequence using the provided [transform] function applied to each pair of elements.
 * The resulting sequence ends as soon as the shortest input sequence ends.
 *
 * The operation is _intermediate_ and _stateless_.
 * @sample samples.collections.Sequences.Transformations.zipWithTransform
 */
public fun <T, R> Sequence<T>.zipWithNext(transform: (a: T, b: T) -> R): Sequence<R> {
    return sequence result
        { val iterator = iterator() if (!iterator.hasNext()) return@result
            var current = iterator.next()
            while (iterator.hasNext()) {
                val next = iterator.next()
                yield(transform(current, next))
                current = next
            }
        }
}

/** * Appends the string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.joinTo
 */
public fun <T, A : Appendable> Sequence<T>.joinTo(buffer: A, separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): A {
    buffer.append(prefix)
    var count = 0
    for (element in this) {
        if (++count > 1) buffer.append(separator)
        if (limit < 0 || count <= limit) buffer.appendElement(element, transform, transform)
    }
    if (limit >= 0 && count > limit) buffer.append(truncated)
    buffer.append(postfix)
    return buffer
}

/** * Creates a string from all the elements separated using [separator] and using the given [prefix] and [postfix] if supplied.
 * If the collection could be huge, you can specify a non-negative value of [limit], in which case only the first [limit] elements will be appended, followed by the [truncated] string (which defaults to "...").
 * The operation is _terminal_.
 * @sample samples.collections.Collections.Transformations.joinToString
 */
public fun <T> Sequence<T>.joinToString(separator: CharSequence = ", ", prefix: CharSequence = "", postfix: CharSequence = "", limit: Int = -1, truncated: CharSequence = "...", transform: ((T) -> CharSequence)? = null): String {
    return joinTo(StringBuilder(), separator, prefix, postfix, limit, truncated, transform).toString()
}

/** * Creates an [Iterable] instance that wraps the original sequence returning its elements when being iterated.
 * The operation is _final_.
 * @sample Open Source U
 */
public fun <T> Sequence<T>.asIterable(): Iterable<T> { return Iterable { this.iterator() }
}

/** * Returns this sequence as a [Sequence].
 * @kotlin.internal.InlineOnly
 */
public inline fun <T> Sequence<T>.asSequence(): Sequence<T>
```kotlin
/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfByte")
public fun Sequence<Byte>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfShort")
public fun Sequence<Short>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfInt")
public fun Sequence<Int>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfLong")
public fun Sequence<Long>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfFloat")
public fun Sequence<Float>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns an average value of elements in the sequence.
 */
@kotlin.jvm.JvmName("averageOfDouble")
public fun Sequence<Double>.average(): Double {
    var sum: Double = 0.0
    var count: Int = 0
    for (element in this) {
        sum += element
        checkCountOverflow(++count)
    }
    return if (count == 0) Double.NaN else sum / count
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfByte")
public fun Sequence<Byte>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfShort")
public fun Sequence<Short>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfInt")
public fun Sequence<Int>.sum(): Int {
    var sum: Int = 0
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfLong")
public fun Sequence<Long>.sum(): Long {
    var sum: Long = 0L
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfFloat")
public fun Sequence<Float>.sum(): Float {
    var sum: Float = 0.0f
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 */
@kotlin.jvm.JvmName("sumOfDouble")
public fun Sequence<Double>.sum(): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += element
    }
    return sum
}
```

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
 * contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("SetsKt")

package kotlin.collections

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib
//

import kotlin.random.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns a set containing all elements of the original
 * set except the given [element].
 * 
 * The returned set preserves the element iteration order of the original set.
 * 
 * @param element the element to remove
 * @return a set with all elements of the original set except the given [element]
 * 
 * The operation is _terminal_.
 */
public operator fun <T> Set<T>.minus(element: T): Set<T> {
    var val result = LinkedHashSet<T>(mapCapacity(size))
    var removed = false
    return this.filterTo(result) { if (!removed && &
```
== element) { removed = true; false } else true }

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the given [element].
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(element: T): Set<T> {
    return plusSetWith(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    return plusSetWith(elements)
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {
    return plusSetWith(elements)
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    return plusSetWith(elements)
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the given [element].
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] array may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Array<out T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] collection may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Iterable<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the elements contained in the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 * The [elements] sequence may be converted to a [HashSet] to speed up the operation, thus the elements are required to have a correct and stable implementation of `hashCode()` that doesn't change between successive invocations.
 */
public operator fun <T> Set<T>.minus(elements: Sequence<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.removeAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set except the given [element].
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.minusElement(element: T): Set<T> {
    return minus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [element] if it isn't already in this set.
 * The returned set preserves the element iteration order of the original set.
 */
@kotlin.internal.InlineOnly
public inline fun <T> Set<T>.plusElement(element: T): Set<T> {
    return plus(element)
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] array.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Array<out T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] collection.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Iterable<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}

/**
 * Returns a set containing all elements of the original set and then the given [elements] sequence.
 * The returned set preserves the element iteration order of the original set.
 */
public operator fun <T> Set<T>.plus(elements: Sequence<T>): Set<T> {
    var result = LinkedHashSet<T>(this)
    result.addAll(elements)
    return result
}
CharSequence.elementAt(index: Int): Char

/**
 * Returns a character at the given [index] or the result of calling the
 * [defaultValue] function if the [index] is out of bounds of this char sequence.
 */
public fun CharSequence.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns a character at the given [index] or `null` if the [index] is out of bounds of this char sequence.
 */
public fun CharSequence.getOrNull(index: Int): Char? {
    return if (index >= 0 && index <= lastIndex) get(index) else null
}

/**
 * Returns index of the first character matching the given [predicate], or -1 if the char sequence does not contain such character.
 */
public inline fun CharSequence.indexOfFirst(predicate: (Char) -> Boolean): Int {
    for (index in indices) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns index of the last character matching the given [predicate], or -1 if the char sequence does not contain such character.
 */
public inline fun CharSequence.indexOfLast(predicate: (Char) -> Boolean): Int {
    for (index in indices.reversed()) {
        if (predicate(this[index])) {
            return index
        }
    }
    return -1
}

/**
 * Returns the first character matching the given [predicate], or `null` if there is no such character.
 */
public fun CharSequence.first(): Char {
    if (isEmpty())
        throw NoSuchElementException("Char sequence is empty.")
    return this[0]
}

/**
 * Returns the first character matching the given [predicate].
 */
public inline fun CharSequence.firstOrNull(): Char? {
    return if (isEmpty()) null else this[0]
}

/**
 * Returns the first character matching the given [predicate] and, if no such character was found, the result of calling the [defaultValue] function.
 */
public fun CharSequence.getOrElse(index: Int): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else this[index]
}

/**
 * Returns the first character matching the given [predicate] and, if no such character was found, the result of calling the [defaultValue] function.
 */
public fun CharSequence.getOrElse(index: Int, defaultValue: (Int) -> Char): Char {
    return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index)
}

/**
 * Returns a random character from this char sequence.
 */
@SinceKotlin("1.3")
public inline fun CharSequence.random(): Char {
    return CharRange(0, CharRange.MAX).random()
}
return random(Random)n\n/*\n * returns a random character from this char sequence using the specified source of randomness.
 * @throws NoSuchElementException if this char sequence is empty.
 */
@SinceKotlin("1.3")
public fun CharSequence.random(random: Random): Char {
    return randomOrNull(Random)
}

/**
 * Returns a random character from this char sequence using the specified source of randomness, or `null` if this char sequence is empty.
 * @throws NoSuchElementException if this char sequence is empty.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public inline fun CharSequence.randomOrNull(random: Random): Char? {
    if (isEmpty())
        return null
    return get(random.nextInt(length))
}

/**
 * Returns the single character matching the given [predicate], or throws exception if there is no or more than one matching character.
 * @throws NoSuchElementException if this char sequence is empty or has more than one character.
 */
public fun CharSequence.single(): Char {
    return when (length) {
        0 -> throw NoSuchElementException("Char sequence is empty.")
        1 -> this[0]
        else -> throw NoSuchElementException("Char sequence contains more than one element.")
    }
}

/**
 * Returns the single character matching the given [predicate], or `null` if there is no or more than one matching character.
 * @throws NoSuchElementException if this char sequence is empty or has more than one character.
 */
public inline fun CharSequence.singleOrNull(predicate: (Char) -> Boolean): Char? {
    var single: Char? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Char sequence contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a subsequence of this char sequence with the first n characters removed.
 * @throws IllegalArgumentException if n is negative.
 * @sample samples.text.Strings.drop
 */
public fun CharSequence.drop(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(n.coerceAtMost(length), length)
}

/**
 * Returns a string with the first n characters removed.
 * @throws IllegalArgumentException if n is negative.
 * @sample samples.text.Strings.drop
 */
public fun String.drop(n: Int): String {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return substring(n.coerceAtMost(length))
}

/**
 * Returns a subsequence of this char sequence containing all characters except last characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun CharSequence.dropLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return subSequence(0, index + 1)
    return emptyCharSequence
}

/**
 * Returns a string containing all characters except last characters that satisfy the given [predicate].
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropLastWhile(predicate: (Char) -> Boolean): String {
    for (index in lastIndex downTo 0)
        if (!predicate(this[index]))
            return substring(0, index + 1)
    return emptyString
}
/**
 * Returns a string containing all characters except first characters that satisfy the given [predicate].
 * 
 * @sample samples.text.Strings.drop
 */
public inline fun String.dropWhile(predicate: (Char) -> Boolean): String {
    for (index in this.indices)
        if (!predicate(this[index]))
            return substring(index)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that
 * match the given [predicate].
 */
public inline fun CharSequence.dropWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)
    return ""
}

/**
 * Returns a char sequence containing only those characters from the original char sequence that
 * do not match the given [predicate].
 */
public inline fun CharSequence.filterNot(predicate: (Char) -> Boolean): CharSequence {
    return filterNotTo(StringBuilder(), predicate)
}

/**
 * Returns a string containing only those characters from the original string that do not match the given [predicate].
 */
public inline fun String.filterNot(predicate: (Char) -> Boolean): String {
    return filterNotTo(StringBuilder(), predicate).toString()
}

/**
 * Returns a char sequence containing characters of the original char sequence at the
 * specified range of [indices].
 */
public fun CharSequence.slice(indices: IntRange): CharSequence {
    if (indices.isEmpty()) return ""
    return subSequence(indices)
}

/**
 * Returns a string containing characters of the original string at the specified range of [indices].
 */
public fun String.slice(indices: IntRange): String {
    if (indices.isEmpty()) return ""
    return substring(indices)
}

/**
 * Returns a char sequence containing characters of the original char sequence at specified 
 * [indices].
 */
public fun CharSequence.slice(indices: Iterable<Int>): CharSequence {
    val size = indices.collectionSizeOrDefault(10)
    if (size == 0) return ""
    val result = StringBuilder(size)
    for (i in indices) {
        result.append(get(i))
    }
    return result
}

/**
 * Returns a string containing characters of the original string at specified [indices].
 */
@kotlin.internal.InlineOnly
public inline fun String.slice(indices: Iterable<Int>): String {
    val result = StringBuilder(indices.size)
    for (i in indices) {
        result.append(get(i))
    }
    return result
}

/**
 * Returns a subsequence of this char sequence containing the
 * first [n] characters from this char sequence, or the entire char sequence if this char sequence is shorter.
 */
public fun CharSequence.take(n: Int): CharSequence {
    require(n >= 0) { "Requested character count $n is less than zero." }
    return subSequence(0, n)
}
### Open Source Used In IOx-FogDirector-AC18 1.17.0

2204

```kotlin
/**
   * Returns a string containing the first [n] characters from this string.
   * If this string is shorter, returns the entire string.
   * @param length The length of the substring.
   * @return The substring.
   */
public fun String.substring(length: Int): String {
    return subSequence(0, length)
}

/**
   * Returns an empty subsequence of this char sequence.
   * If this char sequence is shorter, returns the entire char sequence.
   * @return An empty subsequence.
   */
public fun CharSequence.EMPTY_SUBSEQUENCE: CharSequence {
    return subSequence(0, length)
}

/**
   * Returns a subsequence of this char sequence containing the last [n] characters from this char sequence.
   * If this char sequence is shorter, returns the entire char sequence.
   * @param length The length of the substring.
   * @return The substring.
   */
public fun CharSequence.takeLast(length: Int): CharSequence {
    return subSequence(length - 1, length)
}

/**
   * Returns a subsequence of this char sequence containing the last [n] characters.
   * If this char sequence is shorter, returns the entire char sequence.
   * @param length The length of the substring.
   * @return The substring.
   */
public fun CharSequence.takeLastWhile(predicate: (Char) -> Boolean): CharSequence {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return subSequence(index + 1, length)
        }
    }
    return subSequence(0, length)
}

/**
   * Returns a character sequence with characters in reversed order.
   * @return A reversed character sequence.
   */
public fun String.reversed(): String {
    return when (this as CharSequence).isNotEmpty.run { reverse() } ?: emptySequence().toString()
}

/**
   * Returns a map containing key-value pairs provided by [transform] function applied to characters of the given char sequence.
   * If any two characters would have the same key returned by [keySelector] function applied to each character, the last one gets added to the map.
   * @param capacity The capacity of the map.
   * @return A map.
   */
public inline fun <K, V> CharSequence.associate(transform: (Char) -> Pair<K, V>): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateTo(LinkedHashMap<K, V>(capacity), transform)
}
```

---

```kotlin
/**
   * Returns a map containing key-value pairs provided by [valueTransform] function applied to characters of the given char sequence.
   * If any two characters would have the same key returned by [keySelector] function applied to each character, the last one gets added to the map.
   * @param capacity The capacity of the map.
   * @return A map.
   */
public inline fun <K, V> CharSequence.associateBy(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

---

```kotlin
/**
   * Returns a map containing key-value pairs provided by [valueTransform] function applied to characters of the given char sequence.
   * If any two characters would have the same key returned by [keySelector] function applied to each character, the last one gets added to the map.
   * @param capacity The capacity of the map.
   * @return A map.
   */
public inline fun <K, V> CharSequence.associateByWithValueTransform(keySelector: (Char) -> K, valueTransform: (Char) -> V): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByWithValueTransformTo(LinkedHashMap<K, V>(capacity), keySelector, valueTransform)
}
```

---

```kotlin
/**
   * Returns a map containing key-value pairs provided by [valueTransform] function applied to characters of the given char sequence.
   * If any two characters would have the same key returned by [keySelector] function applied to each character, the last one gets added to the map.
   * @param capacity The capacity of the map.
   * @return A map.
   */
public inline fun <K, V> CharSequence.associateByWith(keySelector: (Char) -> K): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByTo(LinkedHashMap<K, V>(capacity), keySelector)
}
```

---

```kotlin
/**
   * Returns a map containing key-value pairs provided by [valueTransform] function applied to characters of the given char sequence.
   * If any two characters would have the same key returned by [keySelector] function applied to each character, the last one gets added to the map.
   * @param capacity The capacity of the map.
   * @return A map.
   */
public inline fun <K, V> CharSequence.associateByWithWithValueTransform(keySelector: (Char) -> K): Map<K, V> {
    val capacity = mapCapacity(length).coerceAtLeast(16)
    return associateByWithValueTransformTo(LinkedHashMap<K, V>(capacity), keySelector)
}
```
= mapCapacity(length).coerceAtLeast(16)/n    return associateByTo(LinkedHashMap<K, V>(capacity),
keySelector, valueTransform)/n\n\n** Populates and returns the [destination] mutable map with key-value pairs,\n* where key is provided by the [keySelector] function and\n* value is the character itself.\n* If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n* @sample samples.text.Strings.associateByTo\n\n** Populates and returns the [destination] mutable map with key-value pairs,\n* where key is provided by the [keySelector] function and\n* and value is provided by the [valueTransform] function applied to each character of the given char sequence.\n* If any two characters would have the same key returned by [keySelector] the last one gets added to the map.\n* @sample samples.text.Strings.associateByToWithValueTransform\n\n** Populates and returns the [destination] mutable map with key-value pairs,\n* provided by [transform] function applied to each character of the given char sequence.\n* If any two characters are equal, the last one gets added to the map.\n* @sample samples.text.Strings.associateTo\n\n** Returns a [Map] where keys are characters from the given char sequence and values are\n* produced by the [valueSelector] function applied to each character.\n* The returned map preserves the entry iteration order of the original char sequence.\n* @sample samples.text.Strings.associateWith\n\n** Returns a [HashSet] of all characters.\n* The returned set preserves the element iteration order of the original char sequence.\n* @sample samples.collections.Collections.Transformations.flatMap\n\n** Returns a single list of all elements yielded from results of [transform] function being invoked on each character of original char sequence.\n* @sample samples.collections.Collections.Transformations.flatMap
val list = transform(element)\n    destination.addAll(list)\n  return destination\n}

Groups characters of the original char sequence by the key returned by the given [keySelector] function applied to each character and returns a map where each group key is associated with a list of corresponding characters. The returned map preserves the entry iteration order of the keys produced from the original char sequence.

CharSequence.groupBy(keySelector: (Char) -> K): Map<K, List<Char>>

Groups values returned by the [valueTransform] function applied to each character of the original char sequence by the key returned by the given [keySelector] function and returns a map where each group key is associated with a list of corresponding values. The returned map preserves the entry iteration order of the keys produced from the original char sequence.

CharSequence.groupByTo(destination: M, keySelector: (Char) -> K, valueTransform: (Char) -> V):

CharSequence.groupingBy(crossinline keySelector: (Char) -> K):

Groups characters of the original char sequence by the key returned by the given [keySelector] function applied to each character and puts to the [destination] map each group key associated with a list of corresponding characters. The returned map preserves the entry iteration order of the keys produced from the original char sequence.

CharSequence.groupingByTo(destination: M, keySelector: (Char) -> K):

CharSequence.map(transform: (Char) -> R):

CharSequence.mapIndexed(transform: (index: Int, Char) -> R):

CharSequence.mapIndexedNotNull(transform: (index: Int, Char) -> R?):

CharSequence.mapIndexedNotNonNull(transform: (index: Int, Char) -> R?):

Open Source Used In IOx-FogDirector-AC18 1.17.0 2206
mutable<in R>> CharSequence.mapIndexedNotNullTo(destination: C, transform: (index: Int, Char) -> R?): C {
    forEachIndexed { index, element -> transform(index, element)?.let { destination.add(it) } }
    return destination
}

/**
 * Applies the given [transform] function to each character and its index in the original char sequence\n and appends the results to the given [destination].\n * @param [transform] function that takes the index of a character and the character itself\n and returns the result of the transform applied to the character.\n */
public inline fun <R, C : MutableCollection<in R>> CharSequence.mapIndexedTo(destination: C, transform: (index: Int, Char) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++,
            item))
    return destination
}

/**
 * Returns a list containing only the non-null results of applying the given
 * [transform] function\n to each character in the original char sequence\n * and appends only the non-null results to the given [destination].\n */
public inline fun <R : Any>
    CharSequence.mapNotNull(transform: (Char) -> R?): List<R> {
    return mapNotNullTo(ArrayList<R>(),
        transform)
}

/**
 * Applies the given [transform] function\n to each character in the original char sequence
 * and appends only the non-null results to the given [destination].\n */
public inline fun <R : Any, C : MutableCollection<in R>> CharSequence.mapNotNullTo(destination: C, transform: (Char) -> R?): C {
    forEach { element -> transform(element)?.let { destination.add(it) } }
    return destination
}

/**
 * Applies the given [transform] function\n to each character of the original char sequence
 * and appends the results to the given [destination].\n * @param [predicate] function that takes the index of a character
 * and returns true if the character matches the given [predicate].\n */
public inline fun <R> CharSequence.fold(initial: R, operation: (acc: R, Char) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying \n * [operation] from left to right to each character and current accumulator value
 * and each character with its index in the original char sequence.\n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n */
public inline fun <R> CharSequence.foldIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): R {
    var index = 0
    var accumulator = initial
    for (item in this) accumulator = operation(index++,
        accumulator, item)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying \n * [operation] from right to left to each character and current accumulator value
 * and each character with its index in the original char sequence.\n * @param [operation] function that takes the index of a character, current accumulator value\n * and the character itself, and calculates the next accumulator value.\n */
public inline fun <R> CharSequence.foldRight(initial: R, operation: (Char, acc: R) -> R): R {
    var index = lastIndex
    var accumulator = initial
    while (index >= 0) { accumulator = operation(get(index--), accumulator) }
    return accumulator
}
operation: (index: Int, Char, acc: R) -> R: R \( \{ \text{ var index = lastIndex}\text{ var accumulator = initial while (index >= 0) \{\text{ accumulator = operation(index, get(index), accumulator)\text{ index --}\}\text{ return accumulator}\}\text{ returnAccumulator}\text{ returnAccumulator}\}\text{ Performs the given [action] on each character.}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\}\text{ Performs the given [action] on each character, providing sequential index with the character.}\text{ returnAccumulator}\text{ @param [action] function that takes the index of a character and the character itself and performs the desired action on the character.}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\}\text{ returnAccumulator}\}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator}\text{ returnAccumulator\n
```kotlin
/*\npublic inline fun CharSequence.reduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): Char {\n  if (isEmpty()) throw UnsupportedOperationException("Empty char sequence can't be reduced.")\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(index, accumulator, this[index])\n  }\n  return accumulator\n}\n
Accumulates value starting with the first character and applying [operation] from left to right to current accumulator value and each character. Returns null if the char sequence is empty.\n*/

@SinceKotlin("1.3")\npublic inline fun CharSequence.reduceOrNull(operation: (acc: Char, Char) -> Char): Char? {
  if (isEmpty()) return null\n  var accumulator = this[0]\n  for (index in 1..lastIndex) {\n    accumulator = operation(accumulator, this[index])\n  }\n  return accumulator\n}

/**
 * Accumulates value starting with last character and applying [operation] from right to left to each character and
 * current accumulator value.\n */
public inline fun CharSequence.reduceRight(operation: (Char, acc: Char) -> Char): Char {
  var index = lastIndex\n  if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")\n  var accumulator = get(index--)\n  while (index >= 0) {\n    accumulator = operation(get(index), accumulator)\n  }\n  return accumulator\n}

/**
 * Accumulates value starting with last character and applying [operation] from right to left to each character and
 * current accumulator value.\n */
public inline fun CharSequence.reduceRightIndexed(operation: (index: Int, Char, acc: Char) -> Char): Char {
  var index = lastIndex\n  if (index < 0) throw UnsupportedOperationException("Empty char sequence can't be reduced.")\n  var accumulator = get(index--)\n  while (index >= 0) {\n    accumulator = operation(index, get(index), accumulator)\n    --index\n  }\n  return accumulator\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each character and current accumulator value. Returns null if the char sequence is empty.\n */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic inline fun CharSequence.scan(initial: R, operation: (acc: R, Char) -> R): List<R> {
  if (isEmpty()) return listOf(initial)\n  val result = ArrayList<R>(length + 1).apply { add(initial) }\n  var accumulator = initial\n  for (element in this) {\n    accumulator = operation(accumulator, element)\n    result.add(accumulator)\n  }\n  return result\n}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each character, its index in the original char sequence and current accumulator value that starts with [initial] value.\n */
@SinceKotlin("1.3")\n@ExperimentalStdlibApi\npublic inline fun CharSequence.scanIndexed(initial: R, operation: (index: Int, acc: R, Char) -> R): List<R> {
  if (isEmpty()) return listOf(initial)\n  val result = ArrayList<R>(length + 1).apply { add(initial) }\n  var accumulator = initial\n  for (element in this) {\n    accumulator = operation(accumulator, element, index)\n    result.add(accumulator)\n  }\n  return result\n}
```

---

**Open Source Used In IOx-FogDirector-AC18 1.17.0 2209**
values generated by applying [operation] from left to right to each character and current accumulator value that starts with the first character of this char sequence.

Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

@sinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduce(operation: (acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply {
        add(accumulator)
    }
    for (index in 1 until length) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right to each character, its index in the original char sequence and current accumulator value that starts with the first character of this char sequence.
 * 
 * Note that 'acc' value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * 
 * @param operation function that takes the index of a character, current accumulator value and the character itself, and calculates the next accumulator value.
 * 
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@sinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun CharSequence.scanReduceIndexed(operation: (index: Int, acc: Char, Char) -> Char): List<Char> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<Char>(length).apply {
        add(accumulator)
    }
    for (index in 1 until length) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
public inline fun CharSequence.sumBy(selector: (Char) -> Int): Int {
    var sum: Int = 0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Returns the sum of all values produced by [selector] function applied to each character in the char sequence.
 */
public inline fun CharSequence.sumByDouble(selector: (Char) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
 * Splits this char sequence into a list of strings each not exceeding the given [size].
 * 
 * The last string in the resulting list may have less characters than the given [size].
 * 
 * @param size the number of elements to take in each string, must be positive and can be greater than the number of elements in this char sequence.
 * 
 * @sample samples.collections.Collections.Transformations.chunked
 */
@sinceKotlin("1.2")
public fun CharSequence.chunked(size: Int): List<String> {
    return windowed(size, size, partialWindows = true)
}

/**
 * Splits this char sequence into several char sequences each not exceeding the given [size] and applies the given [transform] function to an each.
 * 
 * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.
 * You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * 
 * The last char sequence may have less characters than the given [size].
 * 
 * @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.
 * 
 * @sample samples.text.Strings.chunkedTransform
 */
@sinceKotlin("1.2")
public fun <R> CharSequence.chunked(size: Int, transform: (CharSequence) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

/**
 * Splits this char sequence into a sequence of strings each not exceeding the given [size].
 * 
 * The last string in the resulting sequence may have less characters than the given [size].
 * 
 * @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.
 * 
 * @sample samples.collections.Collections.Transformations.chunkedSequence
 */
@sinceKotlin("1.2")
public fun <R> CharSequence.chunkedSequence(size: Int): Sequence<String> {
    return windowed(size, size, partialWindows = true)
}
The last char sequence may have less characters than the given [size].

* @param size the number of elements to take in each char sequence, must be positive and can be greater than the number of elements in this char sequence.

* @param sample * @sample samples.text.Strings.chunkedTransformToSequence

* @public fun <R> CharSequence.chunkedSequence(size: Int, transform: (CharSequence) -> R): Sequence<R> {
    return windowedSequence(size, size, partialWindows = true, transform = transform)
}

* Splits the original char sequence into pair of char sequences, [size] * where *first* char sequence contains characters for which [predicate] yielded 'true', \n * while *second* char sequence contains characters for which [predicate] yielded 'false'.

* @public inline fun CharSequence.partition(predicate: (Char) -> Boolean): Pair<CharSequence, CharSequence> {
    return WindowedSequence(size, size, partialWindows = true, transform = transform)
}

* Splits the original string into pair of strings, where *first* string contains characters for which [predicate] yielded 'true', \n * while *second* string contains characters for which [predicate] yielded 'false'.

* @public inline fun String.partition(predicate: (Char) -> Boolean): Pair<String, String> {
    return WindowedSequence(size, size, partialWindows = true, transform = transform)
}

* Returns a list of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string.

* Several last strings may have less characters than the given [size].

* Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window

* @param step the number of elements to move the window forward by on an each step, by default 1

* @param partialWindows controls whether or not to keep partial windows in the end if any,

* @param sample * @sample samples.collections.Sequences.Transformations.takeWindows

* @public fun CharSequence.windowed(size: Int, step: Int = 1, partialWindows: Boolean = false): List<String> {
    return WindowedSequence(size, size, partialWindows = true, transform = transform)
}

* Returns a list of results of applying the given transform function to each char sequence representing a view over the window of the given [size] sliding along this char sequence with the given [step].

* Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside that function.

* You should not store it or allow it to escape in some way, unless you made a snapshot of it.

* Several last char sequences may have less characters than the given [size].

* Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window

* @param step the number of elements to move the window forward by on an each step, by default 1

* @param partialWindows controls whether or not to keep partial windows in the end if any,

* @param sample * @sample samples.collections.Sequences.Transformations.averageWindows

* @public fun <R> CharSequence.averaged(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): List<R> {
    return windowed(size, size, partialWindows = true, transform = transform)
}

* Returns a sequence of snapshots of the window of the given [size] sliding along this char sequence with the given [step], where each snapshot is a string.

* Several last strings may have less characters than the given [size].

* Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.

* @param size the number of elements to take in each window

* @param step the number of elements to move the window forward by on an each step, by default 1

* @param partialWindows controls whether or not to keep partial windows in the end if any,
Boolean = false): Sequence<String> { return windowedSequence(size, step, partialWindows) { it.toString() }

/**
 * Returns a sequence of results of applying the given [transform] function to
 * each char sequence representing a view over the window of the given [size]
 * sliding along this char sequence with the given [step].
 * Note that the char sequence passed to the [transform] function is ephemeral and is valid only inside
 * that function. You should not store it or allow it to escape in some way, unless you made a snapshot of it.
 * Several last char sequences may have less characters than the given [size].
 * Both [size] and [step] must be positive and can be greater than the number of elements in this char sequence.
 * @param size the number of elements to take in each window
 * @param step the number of elements to move the window forward by on each step, by default 1
 * @param partialWindows controls whether or not to keep partial windows in the end if any,
 * by default `false` which means partial windows won't be preserved
 * @sample samples.collections.Sequences.Transformations.averageWindows
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.windowedSequence(size: Int, step: Int = 1, partialWindows: Boolean = false, transform: (CharSequence) -> R): Sequence<R> {
    checkWindowSizeStep(size, step)
    val windows = (if (partialWindows) indices else 0 until length - size + 1) step step
    return windows.asSequence().map { index ->
        val end = index + size
        val coercedEnd = if (end < 0 || end > length) length else end
        transform(subSequence(index, coercedEnd))
    }

/**
 * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index
 * using the provided [transform] function applied to each pair of characters.
 * The returned list has length of the shortest char sequence.
 * @param other the other char sequence
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {
    val size = length - 1
    if (size < 1) return emptyList()
    val result = ArrayList<R>(size)
    for (index in 0 until size) {
        result.add(transform(this[index], this[index + 1]))
    }
    return result
}

/**
 * Returns a list containing the results of applying the given [transform] function to
 * each char sequence returning its characters when being iterated.
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.zipWithNext(): List<Pair<Char, Char>> {
    return zipWithNext { a, b -> a to b }
}

/**
 * Creates an [Iterable] instance that wraps the original char sequence returning its characters when being iterated,
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.asIterable(): Iterable<Char> {
    if (this is String && isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original char sequence returning its characters when being iterated.
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.asSequence(): Sequence<Char> {
    if (this is String && isEmpty()) return emptyList()
    return Sequence { this.iterator() }
}

/**
 * Returns a list of pairs of each two adjacent characters in this char sequence.
 * The returned list is empty if this char sequence contains less than two characters.
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.zipWithNext(): List<Pair<Char, Char>> {
    return zipWithNext { a, b -> a to b }
}

/**
 * Returns a list of pairs built from the characters of `this` and the [other] char sequences with the same index
 * using the provided [transform] function applied to each pair of characters.
 * The returned list has length of the shortest char sequence.
 * @param other the other char sequence
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun <R> CharSequence.zipWithNext(transform: (a: Char, b: Char) -> R): List<R> {
    val size = length - 1
    if (size < 1) return emptyList()
    val result = ArrayList<R>(size)
    for (index in 0 until size) {
        result.add(transform(this[index], this[index + 1]))
    }
    return result
}

/**
 * Creates an [Iterable] instance that wraps the original char sequence returning its characters when being iterated,
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.asIterable(): Iterable<Char> {
    if (this is String && isEmpty()) return emptyList()
    return Iterable { this.iterator() }
}

/**
 * Creates a [Sequence] instance that wraps the original char sequence returning its characters when being iterated.
 * @sample samples.collections.Collections.Transformations.zipWithNextToFindDeltas
 */
@SinceKotlin("1.2")
public fun CharSequence.asSequence(): Sequence<Char> {
    if (this is String && isEmpty()) return emptyList()
    return Sequence { this.iterator() }
}
```kotlin
else

    endIndex -= 1
}

return subSequence(startIndex, endIndex + 1)
}

/**
 * Returns a string having leading and trailing characters matching the [predicate] removed.
 */
public inline fun String.trim(predicate: (Char) -> Boolean): String =
    (this as CharSequence).trim(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading characters matching the [predicate] removed.
 */
public inline fun CharSequence.trimStart(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices)
        if (!predicate(this[index]))
            return subSequence(index, length)

    return ""
}

/**
 * Returns a string having leading characters matching the [predicate] removed.
 */
public inline fun String.trimStart(predicate: (Char) -> Boolean): String =
    (this as CharSequence).trimStart(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having trailing characters matching the [predicate] removed.
 */
public inline fun CharSequence.trimEnd(predicate: (Char) -> Boolean): CharSequence {
    for (index in this.indices.reversed())
        if (!predicate(this[index]))
            return subSequence(0, index + 1)

    return ""
}

/**
 * Returns a string having trailing characters matching the [predicate] removed.
 */
public inline fun String.trimEnd(predicate: (Char) -> Boolean): String =
    (this as CharSequence).trimEnd(predicate).toString()

/**
 * Returns a sub sequence of this char sequence having leading and trailing whitespace removed.
 */
public fun CharSequence.trim(): CharSequence = trim(Char::isWhitespace)

/**
 * Returns a string having leading and trailing whitespace removed.
 */
@kotlin.internal.InlineOnly
public inline fun String.trim(): String = (this as CharSequence).trim().toString()

/**
 * Returns a sub sequence of this char sequence having leading whitespace removed.
 */
public fun CharSequence.trimStart(): CharSequence = trimStart(Char::isWhitespace)

/**
 * Returns a string having trailing whitespace removed.
 */
@kotlin.internal.InlineOnly
public inline fun String.trimStart(): String = (this as CharSequence).trimStart().toString()

/**
 * Returns a sub sequence of this char sequence having trailing whitespace removed.
 */
public fun CharSequence.trimEnd(): CharSequence = trimEnd(Char::isWhitespace)

/**
 * Returns a string having trailing whitespace removed.
 */
@kotlin.internal.InlineOnly
public inline fun String.trimEnd(): String = (this as CharSequence).trimEnd().toString()

/**
 * Returns a char sequence with content of this char sequence padded at the beginning
 * to the specified [length] with the specified character or space.
 */
public fun CharSequence.padStart(length: Int, padChar: Char = ' '):

```
(this as CharSequence).padStart(length, padChar).toString().

/**
 * Returns a char sequence with content of this char sequence padded at the end to the specified length with the specified character or space.

 * @param length the desired string length.
 * @param padChar the character to pad string with, if it has length less than the specified. Space is used by default.
 * @return Returns a char sequence of length at least [length] consisting of `this` char sequence appended with [padChar] as many times as are necessary to reach that length.
 */
public fun CharSequence.padStart(length: Int, padChar: Char = ' '): CharSequence {
    if (length < 0)
        throw IllegalArgumentException("Desired length $length is less than zero.")
    if (length <= this.length)
        return this

    val sb = StringBuilder(length)
    sb.append(this)
    for (i in 1..(length - this.length))
        sb.append(padChar)
    return sb
}

/**
 * Pads the string to the specified length at the end with the specified character or space.

 * @param length the desired string length.
 * @param padChar the character to pad string with, if it has length less than the specified. Space is used by default.
 * @return Returns a string of length at least [length] consisting of `this` string appended with [padChar] as many times as are necessary to reach that length.
 */
public fun String.padEnd(length: Int, padChar: Char = ' '): String = (this as CharSequence).padEnd(length, padChar).toString()

/**
 * Returns `true` if this nullable char sequence is either `null` or empty.

 * @sample samples.text.Strings.isNullOrEmpty
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrEmpty(): Boolean {
    contract {
        returns(false) implies (this@isNullOrEmpty != null)
    }

    return this == null || this.length == 0
}

/**
 * Returns `true` if this char sequence is empty (contains no characters).

 * @sample samples.text.Strings.stringIsEmpty
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isEmpty(): Boolean = length == 0

/**
 * Returns `true` if this char sequence is not empty.

 * @sample samples.text.Strings.stringIsNotEmpty
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotEmpty(): Boolean = length > 0

/**
 * Returns `true` if this char sequence is not empty and contains some characters except of whitespace characters.

 * @sample samples.text.Strings.stringIsNotBlank
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence.isNotBlank(): Boolean = !isBlank()

/**
 * Returns `true` if this nullable char sequence is either `null` or empty or consists solely of whitespace characters.

 * @sample samples.text.Strings.stringIsNullOrBlank
 */
@kotlin.internal.InlineOnly
public inline fun CharSequence?.isNullOrBlank(): Boolean {
    contract {
        returns(false) implies (this@isNullOrBlank != null)
    }

    return this == null || isBlank()
}

/**
 * Iterator for characters of the given char sequence.

 */
public operator fun CharSequence.iterator(): CharIterator = object : CharIterator() {
    private var index = 0

    public override fun nextChar(): Char = get(index++)

    public override fun hasNext(): Boolean = index < length
}

/** Returns the string if it is not `null`, or the empty string otherwise.

 */
@kotlin.internal.InlineOnly
public inline fun String?.orEmpty(): String = this ?: ""

/**
 * Returns this char sequence if it's not empty or the result of calling [defaultValue] function if the char sequence is empty.

 * @sample samples.text.Strings.stringIfEmpty
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifEmpty(defaultValue: () -> R): R where C : CharSequence, C : R =
    if (isEmpty()) defaultValue() else this

/**
 * Returns this char sequence if it is not empty and doesn't consist solely of whitespace characters, or the result of calling [defaultValue] function otherwise.

 * @sample samples.text.Strings.stringIfBlank
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun <C, R> C.ifBlank(defaultValue: () -> R): R where C : CharSequence =
    if (isBlank()) defaultValue() else this

/**
 * Returns the range of valid character indices for this char sequence.

 */
public val CharSequence.indices: IntRange
    get() = 0..length - 1

/**
 * Returns the index of the last character in the char sequence or -1 if it is empty.

 * @sample samples.text.Strings.stringIfLastIndex
 */
public val CharSequence.lastIndex: Int
    get() = this.length - 1

/**
 * Returns `true` if this CharSequence has Unicode surrogate pair at the specified index.

 * @sample samples.text.Strings.stringHasSurrogatePairAt
 */
@SinceKotlin("1.3")
@kotlin.internal.InlineOnly
public inline fun CharSequence?.hasSurrogatePairAt(index: Int): Boolean =
    index in 0..length - 2 && this[index].isHighSurrogate() && this[index + 1].isLowSurrogate()
isLowSurrogate() \n\n\n\n\n}  \n
/**
 * Returns a substring specified by the given [range] of indices.
 */
public fun String.substring(range: IntRange): String = substring(range.start, range.endInclusive + 1)

/**
 * Returns a subsequence of this char sequence specified by the given [range] of indices.
 */
public fun CharSequence.subSequence(range: IntRange): CharSequence = subSequence(range.start, range.endInclusive + 1)

/**
 * Returns a substring of chars from a range of this char sequence starting at the [startIndex] and ending right before the [endIndex].
 */
public fun CharSequence.substring(startIndex: Int, endIndex: Int): String = subSequence(startIndex, endIndex + 1).toString()

/**
 * Returns a substring of chars at indices from the specified [range] of this char sequence.
 */
public fun CharSequence.substring(range: IntRange): String = subSequence(range.start, range.endInclusive + 1).toString()

/**
 * Returns a substring before the first occurrence of [delimiter].
 */
public fun String.substringBefore(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring before the first occurrence of [delimiter].
 */
public fun String.substringBefore(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 */
public fun String.substringAfter(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

/**
 * Returns a substring after the first occurrence of [delimiter].
 */
public fun String.substringAfter(delimiter: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}

/**
 * Returns a substring before the last occurrence of [delimiter].
 */
public fun String.substringBeforeLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring before the last occurrence of [delimiter].
 */
public fun String.substringBeforeLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(0, index)
}

/**
 * Returns a substring after the last occurrence of [delimiter].
 */
public fun String.substringAfterLast(delimiter: Char, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + 1, length)
}

/**
 * Returns a substring after the last occurrence of [delimiter].
 */
public fun String.substringAfterLast(delimiter: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else substring(index + delimiter.length, length)
}
```kotlin
open fun CharSequence.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): CharSequence {
    if (endIndex < startIndex)
        throw IndexOutOfBoundsException("End index ($endIndex) is less than start index ($startIndex).")
    val sb = StringBuilder()
    sb.appendRange(this, 0, startIndex)
    sb.append(replacement)
    sb.appendRange(this, endIndex, length)
    return sb.toString()
}

open fun String.replaceRange(startIndex: Int, endIndex: Int, replacement: CharSequence): String {
    (this as CharSequence).replaceRange(startIndex, endIndex, replacement).toString()
}

open fun CharSequence.replaceRange(range: IntRange, replacement: CharSequence): CharSequence {
    removeRange(range.start, range.endInclusive + 1)
    replaceRange(range.start, range.endInclusive + 1, replacement)
    return toString()
}

open fun String.replaceRange(range: IntRange, replacement: CharSequence): String {
    (this as CharSequence).replaceRange(range).toString()
}

open fun CharSequence.removeRange(startIndex: Int, endIndex: Int): CharSequence {
    if (endIndex == startIndex)
        return this.subSequence(startIndex, endIndex)
    val sb = StringBuilder(length - (endIndex - startIndex))
    sb.appendRange(this, 0, startIndex)
    sb.appendRange(this, endIndex, length)
    return sb.toString()
}

open fun String.removeRange(startIndex: Int, endIndex: Int): String {
    return subSequence(startIndex, endIndex).toString()
}

open fun CharSequence.removeRange(range: IntRange): CharSequence {
    return removeRange(range.start, range.endInclusive + 1)
}

open fun String.removeRange(range: IntRange): String {
    return substring(range.start, range.endInclusive + 1)
}

open fun CharSequence.removePrefix(prefix: CharSequence): CharSequence {
    if (startsWith(prefix)) {
        return subSequence(prefix.length, length)
    }
    return subSequence(0, length)
}

open fun String.removePrefix(prefix: CharSequence): String {
    if (startsWith(prefix)) {
        return substring(prefix.length)
    }
    return this
}

open fun CharSequence.removeSuffix(suffix: CharSequence): CharSequence {
    if (endsWith(suffix)) {
        return subSequence(0, length - suffix.length)
    }
    return subSequence(0, length)
}

open fun String.removeSuffix(suffix: CharSequence): String {
    if (endsWith(suffix)) {
        return substring(0, length - suffix.length)
    }
    return this
}

when this char sequence starts with the given [prefix] and ends with the given [suffix], returns a new char sequence having both the given [prefix] and [suffix] removed.

otherwise, returns a new char sequence with the same characters.

```
same characters. */
public fun CharSequence.removeSurrounding(prefix: CharSequence, suffix: CharSequence):
    CharSequence {
        if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {
            return subSequence(prefix.length, length - suffix.length)
        } else {
            return subSequence(0, length)
        }
    }

/**
 * Removes from a string both the given [prefix] and [suffix] if and only if it starts with the [prefix]
 * and ends with the [suffix]. Otherwise returns this string unchanged.
 */
public fun String.removeSurrounding(prefix: CharSequence, suffix: CharSequence): String {
    if ((length >= prefix.length + suffix.length) && startsWith(prefix) && endsWith(suffix)) {
        return substring(prefix.length, length - suffix.length)
    } else {
        return this
    }
}

/**
 * When this char sequence starts with and ends with the given [delimiter], returns a new char
 * sequence having this [delimiter] removed both from the start and end.
 */
public fun CharSequence.removeSurrounding(delimiter: CharSequence): CharSequence = removeSurrounding(delimiter, delimiter)

/**
 * Removes part of string before the first occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBefore(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Removes part of string before the first occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBefore(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Removes part of string after the first occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfter(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)
}

/**
 * Removes part of string after the first occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfter(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = indexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}

/**
 * Removes part of string after the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfterLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + 1, length, replacement)
}

/**
 * Removes part of string after the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceAfterLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(index + delimiter.length, length, replacement)
}

/**
 * Removes part of string before the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBeforeLast(delimiter: Char, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}

/**
 * Removes part of string before the last occurrence of given delimiter with the [replacement] string.
 * If the string does not contain the delimiter, returns [missingDelimiterValue] which defaults to the original string.
 */
public fun String.replaceBeforeLast(delimiter: String, replacement: String, missingDelimiterValue: String = this): String {
    val index = lastIndexOf(delimiter)
    return if (index == -1) missingDelimiterValue else replaceRange(0, index, replacement)
}
missingDelimiterValue else replaceRange(0, index, replacement) in public fun String.replace(oldChar: Char, newChar: Char, ignoreCase: Boolean): String

 JVM- and JS-specific public fun String.replace(oldValue: String, newValue: String, ignoreCase: Boolean): String

Returns a new string obtained by replacing each substring of this char sequence that matches the given regular expression with the given replacement. The [replacement] can consist of any combination of literal text and $-substitutions. To treat the replacement string literally escape it with the [kotlin.text.Regex.Companion.escapeReplacement] method.

replace(0, 0, replacement) in public inline fun CharSequence.replace(regex: Regex, replacement: String): String

replace(regex, replacement) in public fun String.replace(regex: Regex, replacement: String): String

replace(regex, noinline transform: (MatchResult) -> CharSequence) in public inline fun CharSequence.replace(regex: Regex, noinline transform: (MatchResult) -> CharSequence): String

replaceFirst(regex, replacement: String) in public inline fun CharSequence.replaceFirst(regex: Regex, replacement: String): String

matches(regex: Regex) in public inline infix fun CharSequence.matches(regex: Regex): Boolean

regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean) in internal fun CharSequence.regionMatchesImpl(thisOffset: Int, other: CharSequence, otherOffset: Int, length: Int, ignoreCase: Boolean): Boolean

startsWith(char: Char, ignoreCase: Boolean = false) in public fun CharSequence.startsWith(char: Char, ignoreCase: Boolean = false): Boolean

endsWith(char: Char, ignoreCase: Boolean = false) in public fun CharSequence.endsWith(char: Char, ignoreCase: Boolean = false): Boolean

startsWith(prefix: CharSequence, ignoreCase: Boolean = false): Boolean

endsWith(suffix: CharSequence, ignoreCase: Boolean = false): Boolean

commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String

longest string `prefix` such that this char sequence and [other] char sequence both start with this prefix, taking care not to split surrogate pairs. "true" if the [other] has no common prefix, returns the empty string. "false" @param ignoreCase 'true' to ignore character case when matching a character. By default 'false'. @sample samples.text.Strings.commonPrefixWith

commonPrefixWith(0, 0, common prefix and suffix) in public fun CharSequence.commonPrefixWith(other: CharSequence, ignoreCase: Boolean = false): String

val shortestLength = minOf(this.length, other.length) when (i < shortestLength && this[i].equals(other[i], ignoreCase = ignoreCase)) { var i = 0 while (i < shortestLength && this[i].equals(other[i], ignoreCase = ignoreCase)) { i++ } if (this.hasSurrogatePairAt(i - 1) || other.hasSurrogatePairAt(i - 1)) { return if (i == this.length && this[i] == other[i]) return suffix } return prefix } return prefix

other.regionMatches(i, this, 0, this.length - i) return this.regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase) return this.regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase) return this.regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase) return this.regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase)

if (ignoreCase && this is String && prefix is String) return this.regionMatches(startIndex, prefix, 0, prefix.length, ignoreCase)

Common prefix and suffix

returns the longest string `prefix` such that this char sequence and [other] char sequence both start with this prefix, taking care not to split surrogate pairs. "true" if this and [other] have no common prefix, returns the empty string. "false" @param ignoreCase 'true' to ignore character case when matching a character. By default 'false'. @sample samples.text.Strings.commonPrefixWith
searching at. The search proceeds backward toward the beginning of the string.

*param ignoreCase `true` to ignore character case when matching a string. By default `false`.

*return A pair of an index of the last occurrence of matched string from [strings] and the string matched or `null` if none of [strings] are found.

To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from the end toward the beginning of this string, and finds at each position the first element in [strings] that matches this string at that position.

*/

public fun CharSequence.findLastAnyOf(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Pair<Int, String>? =
    findAnyOf(strings, startIndex, ignoreCase, last = true)

/**
 * Finds the index of the first occurrence of any of the specified [strings] in this char sequence,

*param ignoreCase `true` to ignore character case when matching a string. By default `false`.

*return An index of the first occurrence of matched string from [strings] or -1 if none of [strings] are found.

To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from the beginning to the end of this string, and finds at each position the first element in [strings] that matches this string at that position.

*/

public fun CharSequence.indexOfAny(strings: Collection<String>, startIndex: Int = 0, ignoreCase: Boolean = false): Int =
    findAnyOf(strings, startIndex, ignoreCase, last = false)?.first ?: -1

/**
 * Finds the index of the last occurrence of any of the specified [strings] in this char sequence,

*param ignoreCase `true` to ignore character case when matching a string. By default `false`.

*return An index of the last occurrence of matched string from [strings] or -1 if none of [strings] are found.

To avoid ambiguous results when strings in [strings] have characters in common, this method proceeds from the end toward the beginning of this string, and finds at each position the first element in [strings] that matches this string at that position.

*/

public fun CharSequence.lastIndexOfAny(strings: Collection<String>, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int =
    findAnyOf(strings, startIndex, ignoreCase, last = true)?.first ?: -1

/**
 * Returns the index within this string of the first occurrence of the specified character, starting from the specified [startIndex].

*param ignoreCase `true` to ignore character case when matching a character. By default `false`.

*return An index of the first occurrence of [char] or -1 if none is found.

*/

public fun CharSequence.indexOf(char: Char, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this !is String)
        indexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeIndexOf(char, startIndex)
}

/**
 * Returns the index within this char sequence of the first occurrence of the specified 

*param ignoreCase `true` to ignore character case when matching a character.

*return An index of the first occurrence of [string] or -1 if none is found.

*/

public fun CharSequence.indexOf(string: String, startIndex: Int = 0, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this !is String)
        indexOf(string, startIndex, length, ignoreCase)
    else
        nativeIndexOf(string, startIndex)
}

/**
 * Returns the index within this char sequence of the last occurrence of the specified character, starting from the specified [startIndex].

*param ignoreCase `true` to ignore character case when matching a character.

*return An index of the last occurrence of [char] or -1 if none is found.

*/

public fun CharSequence.lastIndexOf(char: Char, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this !is String)
        lastIndexOfAny(charArrayOf(char), startIndex, ignoreCase)
    else
        nativeLastIndexOf(char, startIndex)
}

/**
 * Returns the index within this char sequence of the last occurrence of the specified 

*param ignoreCase `true` to ignore character case when matching a character.

*return An index of the last occurrence of [string] or -1 if none is found.

*/

public fun CharSequence.lastIndexOf(string: String, startIndex: Int = lastIndex, ignoreCase: Boolean = false): Int {
    return if (ignoreCase || this !is String)
        lastIndexOf(string, startIndex, 0, ignoreCase, last = true)
    else
        nativeLastIndexOf(string, startIndex, 0, ignoreCase, last = true)
}
starIndex) * Returns `true` if this char sequence contains the specified [other] sequence of characters as a substring.
* fn * @param ignoreCase `true` to ignore character case when comparing strings. By default `false`.
* fn
* @suppress("INAPPLICABLE_OPERATOR_MODIFIER")
* fn @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Boolean =
    if (other is String)
        other.indexOfChar(ignoreCase) >= 0
    else other.indexOfChar(otherChar, 0, length, ignoreCase) >= 0

* fn
* @Suppress("INAPPLICABLE_OPERATOR_MODIFIER")
* fn @public operator fun CharSequence.contains(char: Char, ignoreCase: Boolean = false): Boolean =
    char.indexOfChar(ignoreCase) >= 0

* fn
* @public operator fun CharSequence.contains(regex: Regex): Boolean =
    regex.containsCharIn(this, 0, this.length, ignoreCase = ignoreCase) > 0

* fn
* @public operator fun CharSequence.contains(regex: Regex): Boolean =
    regex.containsMatchIn(this, 0, this.length, ignoreCase = ignoreCase) > 0

* fn
* @public operator fun CharSequence.contains(char: Char, ignoreCase: Boolean = false): Boolean =
    char.indexOfChar(ignoreCase) >= 0

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)

* fn
* @public operator fun CharSequence.contains(other: CharSequence, ignoreCase: Boolean = false): Sequence<IntRange> =
    other.getRangesDelimitedBy(delimiters, startIndex, limit)
negative, but was $limit." } )
val delimitersList = delimiters.asList() return

DelimitedRangesSequence(this, startIdx, limit, { currentIdx -> findAnyOf(delimitersList, currentIdx, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })\n
\n\nval delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()

return

DelimitedRangesSequence(this, startIndex, limit, { currentIndex -> findAnyOf(delimitersList, currentIndex, ignoreCase = ignoreCase, last = false)?.let { it.first to it.second.length } })

val delimitersList = delimiters.asList()
character sequences: CRLF, LF or CR. * The lines returned do not include terminating line separators.

```kotlin
public fun CharSequence.lineSequence(): Sequence<String> = splitToSequence("\r\n", "\n", "\r")
```

```kotlin
/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences:
 * CRLF, LF or CR. * The lines returned do not include terminating line separators.
 */
public fun CharSequence.lineSequence(): Sequence<String> = splitToSequence("\r\n", "\n", "\r")

```kotlin
/**
 * Splits this char sequence to a list of lines delimited by any of the following character sequences:
 * CRLF, LF or CR. * The lines returned do not include terminating line separators.
 */
public fun
```

```kotlin
@SinceKotlin("1.3")
public actual fun CharArray.concatToString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}
```

```kotlin
/**
 * Concatenates characters in this [CharArray] into a String.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun CharArray.concatToString(): String {
    var result = ""
    for (char in this) {
        result += char
    }
    return result
}
```

```kotlin
/**
 * Concatenates characters in this [CharArray] or its subrange into a String.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun CharArray.concatToString(startIndex: Int = 0, endIndex: Int = this.size): String {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    var result = ""
    for (index in startIndex until endIndex) {
        result += this[index]
    }
    return result
}
```

```kotlin
/**
 * Returns a [CharArray] containing characters of this string.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.toCharArray(): CharArray {
    return CharArray(length) { get(it) }
}
```

```kotlin
/**
 * Returns a [CharArray] containing characters of this string or its substring.
 *
 * @param startIndex the beginning (inclusive) of the substring, 0 by default.
 * @param endIndex the end (exclusive) of the substring, length of this string by default.
 *
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the length
 * of this string.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
@ExperimentalStdlibApi
public actual fun String.toCharArray(startIndex: Int = 0, endIndex: Int = this.length): CharArray {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return CharArray(endIndex - startIndex) {
        get(startIndex + it) }
}
```

```kotlin
/**
 * Decodes a string from the bytes in UTF-8 encoding in this array.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun ByteArray.decodeToString(): String {
    return decodeUtf8(this, 0, size, false)
}
```

```kotlin
/**
 * Decodes a string from the bytes in UTF-8 encoding in this array
 * or its subrange.
 *
 * @param startIndex the beginning (inclusive) of the subrange to decode, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to decode, size of this array by default.
 *
 * @throws IndexOutOfBoundsException if [startIndex] is less than zero or [endIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [startIndex] is greater than [endIndex].
 * @throws CharacterCodingException if the byte array contains malformed UTF-8 byte sequence and
 * [throwOnErrorInvalidSequence] is true.
 */
@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
@ExperimentalStdlibApi
public actual fun
mentalStdlibApi
public actual fun ByteArray.decodeToString(
    startIndex: Int = 0,
    endIndex: Int =
) {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, this.size)
    return decodeUtf8(this, startIndex, endIndex,
    throwOnInvalidSequence)
}

/**
 * Encodes this string to an array of bytes in UTF-8 encoding.
 * Any malformed char sequence is replaced by the replacement byte sequence.
 */
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(): ByteArray =
    encodeUtf8(this, 0, length, false)

/**
 * Encodes this string or its substring to an array of bytes in UTF-8 encoding.
 * @param startIndex the beginning (inclusive) of the substring to encode, 0 by default.
 * @param endIndex the end (exclusive) of the substring to encode, length of this string by default.
 * @param throwOnInvalidSequence specifies whether to throw an exception on malformed char sequence or replace.
 * @throws CharacterCodingException if this string contains malformed char sequence and [throwOnInvalidSequence] is true.
 */
@ExperimentalStdlibApi
public actual fun String.encodeToByteArray(
    startIndex: Int = 0,
    endIndex: Int =
) {
    AbstractList.checkBoundsIndexes(startIndex, endIndex, length)
    return encodeUtf8(this, startIndex, endIndex,
    throwOnInvalidSequence)
}

/**
 * Returns a copy of this string converted to upper case using the rules of the default locale.
 * @sample samples.text.Strings.toUpperCase
 */
public actual inline fun String.toUpperCase(): String = asDynamic().toUpperCase()

/**
 * Returns a copy of this string converted to lower case using the rules of the default locale.
 * @sample samples.text.Strings.toLowerCase
 */
public actual inline fun String.toLowerCase(): String =
    asDynamic().toLowerCase()

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeIndexOf(str:
    String, fromIndex: Int): Int = asDynamic().indexOf(str, fromIndex)

@kotlin.internal.InlineOnly
internal actual inline fun String.nativeLastIndexOf(str:
    String, fromIndex: Int): Int = asDynamic().lastIndexOf(str,
    fromIndex)

@kotlin.internal.InlineOnly
internal inline fun String.nativeStartsWith(s: String, position: Int):
    Boolean = asDynamic().startsWith(s, position)

@kotlin.internal.InlineOnly
internal inline fun String.nativeEndsWith(s: String):
    Boolean = asDynamic().endsWith(s)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int): String =
    asDynamic().substring(startIndex)

@kotlin.internal.InlineOnly
public actual inline fun String.substring(startIndex: Int, endIndex: Int): String =
    asDynamic().substring(startIndex, endIndex)

@kotlin.internal.InlineOnly
public inline fun String.concat(str: String): String =
    asDynamic().concat(str)

@kotlin.internal.InlineOnly
public inline fun String.match(regex: String):
    Array<String>? = asDynamic().match(regex)

private val STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b
    -> a.compareTo(b, ignoreCase = true) }

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String>
    get() =
    STRING_CASE_INSENSITIVE_ORDER

@SinceKotlin("1.3")
@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
@ExperimentalStdlibApi
public actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {
    if (ignoreCase) {
        val n1 = this.length
        val n2 = other.length
        val min = minOf(n1, n2)
        if (min == 0) return n1 - n2
        var start = 0
        while (true) {
            val end = minOf(start + 16, min)
            var s1 = this.substring(start, end)
            var s2 = other.substring(start, end)
            if (s1 != s2) {
                s1 = s1.toUpperCase()
                s2 = s2.toUpperCase()
                if (s1 != s2) {
                    return s1.compareTo(s2)
                }
            }
            if (end == min) break
            start = end
            s1 = other.substring(start, end)
            s2 = this.substring(start, end)
        }
        return n1 - n2
    } else {
        return
    }
}

//native public fun String.trim(): String
//TODO: String.replace to implement effective trimLeading and trimTrailing

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {
    if (ignoreCase) {
        val n1 = this.length
        val n2 = other.length
        val min = minOf(n1, n2)
        if (min == 0) return n1 - n2
        var start = 0
        while (true) {
            val end = minOf(start + 16, min)
            var s1 = this.substring(start, end)
            var s2 = other.substring(start, end)
            if (s1 != s2) {
                s1 = s1.toUpperCase()
                s2 = s2.toUpperCase()
                if (s1 != s2) {
                    return s1.compareTo(s2)
                }
            }
            if (end == min) break
            start = end
            s1 = other.substring(start, end)
            s2 = this.substring(start, end)
        }
        return n1 - n2
    } else {
        return
    }
}

val STRING_CASE_INSENSITIVE_ORDER = Comparator<String> { a, b
    -> a.compareTo(b, ignoreCase = true) }

@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun String.compareTo(other: String, ignoreCase: Boolean = false): Int {
    if (ignoreCase) {
        val n1 = this.length
        val n2 = other.length
        val min = minOf(n1, n2)
        if (min == 0) return n1 - n2
        var start = 0
        while (true) {
            val end = minOf(start + 16, min)
            var s1 = this.substring(start, end)
            var s2 = other.substring(start, end)
            if (s1 != s2) {
                s1 = s1.toUpperCase()
                s2 = s2.toUpperCase()
                if (s1 != s2) {
                    return s1.compareTo(s2)
                }
            }
            if (end == min) break
            start = end
            s1 = other.substring(start, end)
            s2 = this.substring(start, end)
        }
        return n1 - n2
    } else {
        return
    }
}

@SinceKotlin("1.2")
public actual val String.Companion.CASE_INSENSITIVE_ORDER: Comparator<String>
    get() =
    STRING_CASE_INSENSITIVE_ORDER
"*/

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming

Open Source Used In IOx-FogDirector-AC18 1.1.7.0 2224
Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. * ^\n* Auto-generated file. DO NOT EDIT! ^\n
package kotln\n@SinceKotlin("1.3.3")\n@ExperimentalUnsignedTypes\npublic inline class UByteArray\n@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_OF_INLINE_CLASS")\n@PublishedApi\ninternal constructor(@PublishedApi internal val storage: ByteArray) : Collection<UByte> { ^\n
/** Creates a new array of the specified [size], with all elements initialized to zero. */ ^\n
* Returns the array element at the given [index]. This method can be called using the index operator. ^\n* If the [index] is out of bounds of this array, throws an [IndexOutOfBoundsException] except in Kotlin/JS. ^\n** Creates a new array of the specified [size], with all elements initialized to zero. */ ^\n
+** Returns the number of elements in the array. */ ^\n+ Creates an iterator over the elements of the array. */ ^\n+ Create a new array of the specified [size], with all elements initialized to zero. */ ^\n+ Returns the number of elements in the array. */ ^\n+ The function [init] is called for each array element sequentially starting from the first one. */ ^\n+ It should return the value for an array element given its index. */ ^\n+ Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors. */ ^\n+ Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. * / ^\n+ Auto-generated file. DO NOT EDIT! ^\n
package kotln\n@SinceKotlin("1.3.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun UByteArray(size: Int, init: (Int) -> UByte): UByteArray {
    return UByteArray(IntArray(size) { index -> init(index).toByte() })
}

@SinceKotlin("1.3.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\npublic inline fun ubyteArrayOf(vararg elements: UByte): UByteArray = elements

Open Source Used In IOx-FogDirector-AC18 1.17.0 2225
array[index++].toUInt() else throw NoSuchElementException(index.toString())})

  @<br />
  override fun contains(element: UInt): Boolean {
  // TODO: Eliminate this check after KT-30016 gets fixed.
  // Currently JS BE does not generate special bridge method for this method.
  if ((element as Any?) !is UInt)
    return false
  return storage.contains(element.toUInt())
}

  override fun override fun containsAll(elements: Collection<UInt>): Boolean {
    return (elements as Collection<*>).all { it is UInt && storage.contains(it.toInt()) }
  }

  override fun override fun isEmpty(): Boolean = this.storage.size == 0
}

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 * 
 * The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray(size: Int, init: (Int) -> UInt): UIntArray {
    return UIntArray(IntArray(size) { index ->
        init(index).toInt()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun uintArrayOf(vararg elements: UInt): UIntArray = elements

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 * 
 * The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(ULongArray(size) { index ->
        init(index).toULong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 * 
 * The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(ULongArray(size) { index ->
        init(index).toULong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

/**
 * Creates a new array of the specified [size], where each element is calculated by calling the specified
 * [init] function.
 * 
 * The function [init] is called for each array element sequentially starting from the first one.
 * It should return the value for an array element given its index.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray(size: Int, init: (Int) -> ULong): ULongArray {
    return ULongArray(ULongArray(size) { index ->
        init(index).toULong()
    })
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ulongArrayOf(vararg elements: ULong): ULongArray = elements

public inline class ULong @PublishedApi internal
constructor(@PublishedApi internal val data: Long) : Comparable<ULong> {

    companion object {
        /**
         * A constant holding the minimum value an instance of ULong can have.
         */
        public const val MIN_VALUE: ULong = ULong(0)

        /**
         * A constant holding the maximum value an instance of
         * ULong can have.
         */
        public const val MAX_VALUE: ULong = ULong(-1)

        /**
         * The number of bytes used to represent an instance of ULong in a binary form.
         */
        public const val SIZE_BYTES: Int = 8

        /**
         * The number of bits used to represent an instance of ULong in a binary form.
         */
        public const val SIZE_BITS: Int = 64
    }

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if
     * it's less than other, or a positive number if it's greater than other.
     */
    public inline operator fun compareTo(other: UByte): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if
     * it's less than other, or a positive number if it's greater than other.
     */
    public inline operator fun compareTo(other: UShort): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if
     * it's less than other, or a positive number if it's greater than other.
     */
    public inline operator fun compareTo(other: UInt): Int = this.compareTo(other.toULong())

    /**
     * Compares this value with the specified value for order.
     * Returns zero if this value is equal to the specified other value, a negative number if
     * it's less than other, or a positive number if it's greater than other.
     */
    public inline operator fun compareTo(other: ULong): Int = ulongCompare(this.data, other.data)

    /** Adds the other value to this value. */
    public inline operator fun plus(other: UByte): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    public inline operator fun plus(other: UShort): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    public inline operator fun plus(other: UInt): ULong = this.plus(other.toULong())

    /** Adds the other value to this value. */
    public inline operator fun plus(other: ULong): ULong = ULong(this.data.plus(other.data))

    /** Subtracts the other value from this value. */
    public inline operator fun minus(other: UByte): ULong = this.minus(other.toULong())

    /** Subtracts the other value from this value. */
    public inline operator fun minus(other: UShort): ULong = this.minus(other.toULong())

    /** Subtracts the other value from this value. */
    public inline operator fun minus(other: UInt): ULong = this.minus(other.toULong())

    /** Subtracts the other value from this value. */
    public inline operator fun minus(other: ULong): ULong = ULong(this.data.minus(other.data))

    /** Multiplies this value by the other value. */
    public inline operator fun times(other: UByte): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    public inline operator fun times(other: UShort): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    public inline operator fun times(other: UInt): ULong = this.times(other.toULong())

    /** Multiplies this value by the other value. */
    public inline operator fun times(other: ULong): ULong = ULong(this.data.times(other.data))

    /** Divides this value by the other value. */
    public inline operator fun div(other: UByte): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    public inline operator fun div(other: UShort): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    public inline operator fun div(other: UInt): ULong = this.div(other.toULong())

    /** Divides this value by the other value. */
    public inline operator fun div(other: ULong): ULong = ULong(this.data.div(other.data))

    /** Calculates the remainder of dividing this value by the other
       * value. */
    public inline operator fun rem(other: UByte): ULong = this.mod(other.toULong())

    /** Calculates the remainder of dividing this value by the other
       * value. */
    public inline operator fun rem(other: UShort): ULong = this.mod(other.toULong())

    /** Calculates the remainder of dividing this value by the other
       * value. */
    public inline operator fun rem(other: UInt): ULong = this.mod(other.toULong())

    /** Calculates the remainder of dividing this value by the other
       * value. */
    public inline operator fun rem(other: ULong): ULong = ULong(this.data.mod(other.data))

    /** Returns a string representation of this value. */
    override fun toString(): String = String.valueOf(this)

    /** Returns a string representation of this value. */
    override fun hashCode(): Int = this.hashCode()
value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UByte): ULong =
this.rem(other.toULong())
 /** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UShort): ULong =
this.rem(toULong(other))
 /** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: UInt): ULong =
this.rem(toULong(other))
 /** Calculates the remainder of dividing this value by the other value. */
@kotlin.internal.InlineOnly
public inline operator fun rem(other: ULong): ULong =
ulongRemainder(this, other)
 /** Increments this value. */
@kotlin.internal.InlineOnly
public inline operator fun inc(): ULong = ULong(data.inc())
 /** Decrements this value. */
@kotlin.internal.InlineOnly
public inline operator fun dec(): ULong = ULong(data.dec())
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline operator fun and(other: ULong): ULong =
ULong(data and other.data)
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline operator fun or(other: ULong): ULong =
ULong(this.data or other.data)
 /** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline operator fun xor(other: ULong): ULong =
ULong(this.data xor other.data)
 /** Inverts the bits in this value. */
@kotlin.internal.InlineOnly
public inline operator fun inv(): ULong = ULong(data.inv())
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong):
ULong = ULong(data or other.data)
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong):
ULong = ULong(data and other.data)
 /** Shifts this value right by the [bitCount] number of bits. */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong =
ULong(data shr bitCount)
 /** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits
with zeros. */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong =
ULong(data ushr bitCount)
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong =
ULong(this.data and
other.data)
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong =
ULong(this.data or other.data)
 /** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong =
ULong(this.data xor other.data)
 /** Inverts the bits in this value. */
@kotlin.internal.InlineOnly
public inline infix fun inv(): ULong = ULong(data.inv())
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong):
ULong = ULong(data or other.data)
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong):
ULong = ULong(data and other.data)
 /** Shifts this value right by the [bitCount] number of bits. */
@kotlin.internal.InlineOnly
public inline infix fun shr(bitCount: Int): ULong =
ULong(data shr bitCount)
 /** Shifts this value right by the [bitCount] number of bits, filling the leftmost bits
with zeros. */
@kotlin.internal.InlineOnly
public inline infix fun shi(bitCount: Int): ULong =
ULong(data shl bitCount)
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong): ULong =
ULong(this.data and
other.data)
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong): ULong =
ULong(this.data or other.data)
 /** Performs a bitwise XOR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun xor(other: ULong): ULong =
ULong(this.data xor other.data)
 /** Inverts the bits in this value. */
@kotlin.internal.InlineOnly
public inline infix fun inv(): ULong = ULong(data.inv())
 /** Performs a bitwise OR operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun or(other: ULong):
ULong = ULong(data or other.data)
 /** Performs a bitwise AND operation between the two values. */
@kotlin.internal.InlineOnly
public inline infix fun and(other: ULong):
ULong = ULong(data and other.data)
@kotlin.internal.InlineOnly
public inline fun toULong(): ULong = this
/**
 * Converts this [ULong] value to [Float].
 * The resulting value is the closest `Float` to this `ULong` value.
 * In case when this `ULong` value is exactly between two `Float`s,
 * the one with zero at least significant bit of mantissa is selected.
 */
@kotlin.internal.InlineOnly
public inline fun toFloat(): Float = this.toDouble().toFloat()

/**
 * Converts this [ULong] value to [Double].
 * The resulting value is the closest `Double` to this `ULong` value.
 * In case when this `ULong` value is exactly between two `Double`s,
 * the one with zero at least significant bit of mantissa is selected.
 */
@kotlin.internal.InlineOnly
public inline fun toDouble(): Double = ulongToDouble(data)

public override fun toString(): String = ulongToString(data)

/**
 * Converts this [Byte] value to [ULong].
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Byte` value,
 * whereas the most significant 56 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Short] value to [ULong].
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Short` value,
 * whereas the most significant 32 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Int] value to [ULong].
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Int` value,
 * whereas the most significant 32 bits are filled with the sign bit of this value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toULong(): ULong = ULong(this.toLong())

/**
 * Converts this [Long] value to [ULong].
 * If this value is positive, the resulting `ULong` value represents the same numerical value as this `Long` value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toULong(): ULong = ULong(this)

/**
 * Converts this [Float] value to [ULong].
 * The fractional part, if any, is rounded down towards zero.
 * Returns zero if this `Float` value is negative or `NaN`,
 * [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toULong(): ULong = doubleToULong(this.toDouble())

/**
 * Converts this [Double] value to [ULong].
 * The fractional part, if any, is rounded down towards zero.
 * Returns zero if this `Double` value is negative or `NaN`,
 * [ULong.MAX_VALUE] if it's bigger than `ULong.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toULong(): ULong = doubleToULong(this)

public inline fun toInt(): Int
/**
 * Converts this [Byte] value to [Int].
 * Returns zero if this `Byte` value is negative or `NaN`,
 * [Int.MAX_VALUE] if it's bigger than `Int.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Byte.toInt(): Int = Int(this.toLong())

/**
 * Converts this [Short] value to [Int].
 * Returns zero if this `Short` value is negative or `NaN`,
 * [Int.MAX_VALUE] if it's bigger than `Int.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Short.toInt(): Int = Int(this.toLong())

/**
 * Converts this [Int] value to [Int].
 * Returns zero if this `Int` value is negative or `NaN`,
 * [Int.MAX_VALUE] if it's bigger than `Int.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Int.toInt(): Int = Int(this)

/**
 * Converts this [Long] value to [Int].
 * Returns zero if this `Long` value is negative or `NaN`,
 * [Long.MAX_VALUE] if it's bigger than `Long.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Long.toInt(): Int = Int(this)

/**
 * Converts this [Float] value to [Int].
 * Returns zero if this `Float` value is negative or `NaN`,
 * [Float.MAX_VALUE] if it's bigger than `Float.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Float.toInt(): Int = Int(this.toDouble())

/**
 * Converts this [Double] value to [Int].
 * Returns zero if this `Double` value is negative or `NaN`,
 * [Double.MAX_VALUE] if it's bigger than `Double.MAX_VALUE`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Double.toInt(): Int = Int(this)
/** Returns the number of elements in the array. */
public override val size: Int get() = storage.size

/** Creates an iterator over the elements of the array. */
private class Iterator(private val array: ShortArray) : UShortIterator() {
    private var index = 0
    override fun hasNext() = index < array.size
    override fun nextUShort() = if (index < array.size) array[index++].toUShort() else throw NoSuchElementException(index.toString())
}

override fun contains(element: UShort): Boolean {
    // TODO: Eliminate this check after KT-30016 gets fixed.
    // Currently JS BE does not generate special bridge method for this method.
    if ((element as Any?) !is UShort) return false
    return storage.contains(element.toShort())
}

override fun containsAll(elements: Collection<UShort>): Boolean {
    return (elements as Collection<*>).all { it is UShort && storage.contains(it.toShort()) }
}

/** Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray(size: Int, init: (Int) -> UShort): UShortArray {
    return UShortArray(ShortArray(size) { index -> init(index).toShort() })
}

/** Creates a new array of the specified [size], where each element is calculated by calling the specified [init] function. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ushortArrayOf(vararg elements: UShort): UShortArray = elements

/** Returns 1st *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component1(): UInt {
    return get(0)
}

/** Returns 1st *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component1(): ULong {
    return get(0)
}

/** Returns 1st *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component1(): UByte {
    return get(0)
}

/** Returns 1st *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component1(): UShort {
    return get(0)
}

/** Returns 2nd *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntArray.component2(): UInt {
    return get(1)
}

/** Returns 2nd *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.component2(): ULong {
    return get(1)
}

/** Returns 2nd *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.component2(): UByte {
    return get(1)
}

/** Returns 2nd *element* from the array. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.component2(): UShort {
    return get(1)
}
If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.component2(): UByte {
    return get(1)
}

/**
 * Returns 2nd *element* from the array.
 * If the size of this array is less than 2, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.component2(): UShort {
    return get(1)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.component3(): UInt {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.component3(): ULong {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.component3(): UByte {
    return get(2)
}

/**
 * Returns 3rd *element* from the array.
 * If the size of this array is less than 3, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.component3(): UShort {
    return get(2)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.component4(): UInt {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.component4(): ULong {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.component4(): UByte {
    return get(3)
}

/**
 * Returns 4th *element* from the array.
 * If the size of this array is less than 4, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.component4(): UShort {
    return get(3)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.component5(): UInt {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.component5(): ULong {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.component5(): UByte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.component5(): UShort {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UIntArray.component6(): UInt {
    return get(5)
}

/**
 * Returns 6th *element* from the array.
 * If the size of this array is less than 6, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
ULongArray.component6(): ULong {
    return get(5)
}

/**
 * Returns 6th *element* from the array.
 * If the size of this array is less than 6, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UByteArray.component6(): UByte {
    return get(5)
}

/**
 * Returns 6th *element* from the array.
 * If the size of this array is less than 6, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun
UShortArray.component6(): UShort {
    return get(5)
}
public inline operator fun UByteArray.component5(): UByte {
    return get(4)
}

/**
 * Returns 5th *element* from the array.
 * 
 * If the size of this array is less than 5, throws an [IndexOutOfBoundsException] except in Kotlin/JS
 * where the behavior is unspecified.
 */
public inline operator fun UShortArray.component5(): UShort {
    return get(4)
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
Expect fun UIntArray.elementAt(index: Int): UInt

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
Expect fun ULongArray.elementAt(index: Int): ULong

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
Expect fun UByteArray.elementAt(index: Int): UByte

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAt
 */
Expect fun UShortArray.elementAt(index: Int): UShort

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
Expect fun UIntArray.elementAtOrNull(index: Int): UInt?

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
Expect fun ULongArray.elementAtOrNull(index: Int): ULong?

/**
 * Returns an element at the given [index] or `null` if the [index] is out of bounds of this array.
 * 
 * @sample samples.collections.Collections.Elements.elementAtOrNull
 */
Expect fun UByteArray.elementAtOrNull(index: Int): UByte?
samples.collections.Collections.Elements.elementAtOrNull
*\n@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UByteArray.elementAtOrNull(index: Int): UByte? {\n    return this.getOrNull(index)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UShortArray.elementAtOrNull(index: Int): UShort? {\n    return this.getOrNull(index)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UIntArray.find(predicate: (UInt) -> Boolean): UInt? {\n    return firstOrNull(predicate)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
ULongArray.find(predicate: (ULong) -> Boolean): ULong? {\n    return firstOrNull(predicate)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UByteArray.find(predicate: (UByte) -> Boolean): UByte? {\n    return firstOrNull(predicate)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UShortArray.find(predicate: (UShort) -> Boolean): UShort? {\n    return firstOrNull(predicate)\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UIntArray.first(): UInt {\n    return storage.first().toUInt()\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
ULongArray.first(): ULong {\n    return storage.first().toULong()\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UByteArray.first(): UByte {\n    return storage.first().toUByte()\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UShortArray.first(): UShort {\n    return storage.first().toUShort()\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
UIntArray.first(predicate: (UInt) -> Boolean): UInt {\n    for (element in this) if (predicate(element)) return
    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n/**
@SinceKotlin("1.3")\n@ExperimentalUnsignedTypes\n@kotlin.internal.InlineOnly\n@public inline fun
ULongArray.first(predicate: (ULong) -> Boolean): ULong {\n    for (element in this) if (predicate(element)) return
    throw NoSuchElementException("Array contains no element matching the predicate.")\n}\n*/

Open Source Used In IOx-FogDirector-AC18 1.17.0 2233
element is found.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun ULongArray.first(predicate: (ULong) -> Boolean): ULong {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
} * @throws NoSuchElementException if no such element is found.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.first(predicate: (UByte) -> Boolean): UByte {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
} * @throws NoSuchElementException if no such element is found.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UShortArray.first(predicate: (UShort) -> Boolean): UShort {
    for (element in this) if (predicate(element)) return element
    throw NoSuchElementException("Array contains no element matching the predicate.")
} * @throws NoSuchElementException if no such element is found.

// ...
UByteArray.getOrElse(index: Int, defaultValue: (Int) -> UByte): UByte |
| return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) |

UShortArray.getOrElse(index: Int, defaultValue: (Int) -> UShort): UShort |
| return if (index >= 0 && index <= lastIndex) get(index) else defaultValue(index) |

UIntArray.getOrNull(index: Int): UInt? |
| return if (index >= 0 && index <= lastIndex) get(index) else null |

ULongArray.getOrNull(index: Int): ULong? |
| return if (index >= 0 && index <= lastIndex) get(index) else null |

UByteArray.getOrNull(index: Int): UByte? |
| return if (index >= 0 && index <= lastIndex) get(index) else null |

UShortArray.getOrNull(index: Int): UShort? |
| return if (index >= 0 && index <= lastIndex) get(index) else null |

UIntArray.indexOf(element: UInt): Int |
| return storage.indexOf(element.toInt()) |

ULongArray.indexOf(element: ULong): Int |
| return storage.indexOf(element.toLong()) |

UByteArray.indexOf(element: UByte): Int |
| return storage.indexOf(element.toByte()) |

UShortArray.indexOf(element: UShort): Int |
| return storage.indexOf(element.toShort()) |

UIntArray.indexOfFirst(predicate: (UInt) -> Boolean): Int |
| return storage.indexOfFirst { predicate(it.toUInt()) } |

ULongArray.indexOfFirst(predicate: (ULong) -> Boolean): Int |
| return storage.indexOfFirst { predicate(it.toULong()) } |

UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int |
| return storage.indexOfLast { predicate(it.toUByte()) } |

UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int |
| return storage.indexOfLast { predicate(it.toUSHort()) } |
such element.

*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.indexOfLast(predicate: (ULong) -> Boolean): Int {
    return storage.indexOfLast {
        predicate(it.toULong())
    }
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the
 * array does not contain such element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.indexOfLast(predicate: (UByte) -> Boolean): Int {
    return storage.indexOfLast {
        predicate(it.toUByte())
    }
}

/**
 * Returns index of the last element matching the given [predicate], or -1 if the
 * array does not contain such element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.indexOfLast(predicate: (UShort) -> Boolean): Int {
    return storage.indexOfLast {
        predicate(it.toUShort())
    }
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the
 * array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.last(): UInt {
    return storage.last().toUInt()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.last(): ULong {
    return storage.last().toULong()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if the array is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.last(): UByte {
    return storage.last().toUByte()
}

/**
 * Returns the last element.
 * @throws [NoSuchElementException] if no such element is found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.last(): UShort {
    return storage.last().toUShort()
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.lastIndexOf(element: UInt): Int {
    return storage.lastIndexOf(element.toInt())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.lastIndexOf(element: ULong): Int {
    return storage.lastIndexOf(element.toLong())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf(element.toUByte())
}

/**
 * Returns last index of [element], or -1 if the array does not contain element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.lastIndexOf(element: UShort): Int {
    return storage.lastIndexOf(element.toUShort())
}
Returns last index of [element], or -1 if the array does not contain element.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.lastIndexOf(element: UByte): Int {
    return storage.lastIndexOf(element.toByte())
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.lastIndexOf(element: UShort): Int {
    return storage.lastIndexOf(element.toShort())
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.lastOrNull(): UInt? {
    return if (isEmpty()) null else this[size - 1]
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.lastOrNull(): ULong? {
    return if (isEmpty()) null else this[size - 1]
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.lastOrNull(): UByte? {
    return if (isEmpty()) null else this[size - 1]
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.lastOrNull(): UShort? {
    return if (isEmpty()) null else this[size - 1]
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.random(): UInt {
    return random(Random)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.random(): ULong {
    return random(Random)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.random(): UByte {
    return random(Random)
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.random(): UShort {
    return random(Random)
}
```

Returns the last element, or `null` if the array is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.random(random: Random): UInt {
    if (isEmpty())
        throw NoSuchElementException("Array is empty.")
    return get(random.nextInt(size))
}
```

Returns the last element matching the given [predicate], or `null` if no such element was found.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.random(): UInt {
    return random(Random)
}
```

Returns a random element from this array using the specified source of randomness. Returns a random element from this array using the specified source of randomness.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.random(): UByte {
    return random(Random)
}
```

Returns the last element matching the given [predicate], or `null` if no such element was found.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.random(): UInt {
    return random(Random)
}
```

Returns a random element from this array.
*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun ULongArray.random(random: Random): ULong\n\n{\n    if (isEmpty()) throw NoSuchElementException("Array is empty.")\n}\nreturn get(random.nextInt(size))\n* Returns a random element from this array using the specified source of randomness. \n\n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UByteArray.random(random: Random): UByte\n{\n    if (isEmpty()) throw NoSuchElementException("Array is empty.")\n}\nreturn get(random.nextInt(size))\n* Returns a random element from this array using the specified source of randomness. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UShortArray.random(random: Random): UShort\n{\n    if (isEmpty()) throw NoSuchElementException("Array is empty.")\n}\nreturn get(random.nextInt(size))\n* Returns a random element from this array using the specified source of randomness. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UIntArray.randomOrNull(): UInt? \n{\n    return randomOrNull(Random)\n}\n* Returns a random element from this array, or `null` if this array is empty. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun ULongArray.randomOrNull(): ULong? \n{\n    return randomOrNull(Random)\n}\n* Returns a random element from this array, or `null` if this array is empty. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UByteArray.randomOrNull(): UByte? \n{\n    return randomOrNull(Random)\n}\n* Returns a random element from this array, or `null` if this array is empty. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UShortArray.randomOrNull(): UShort? \n{\n    return randomOrNull(Random)\n}\n* Returns a random element from this array, or `null` if this array is empty. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UIntArray.single(): UInt \n{\n    return storage.single().toUInt()\n}\n* Returns the single element, or throws an exception if the array is empty or has more than one element. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun ULongArray.single(): ULong \n{\n    return storage.single().toULong()\n}\n* Returns the single element, or throws an exception if the array is empty or has more than one element. \n\n*\@SinceKotlin("1.3")*\@ExperimentalUnsignedTypes*\@public fun UByteArray.single(): UByte \n{\n    return storage.single().toUByte()\n}\n* Returns the single element, or throws an exception if the array is empty or has more than one element. \n
---

Open Source Used In IOx-FogDirector-AC18 1.17.0 2238
public inline fun UShortArray.single(): UShort {
    return storage.single().toUShort()}

/**
 * Returns the single element
 * matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun UIntArray.single(predicate: (UInt) -> Boolean): UInt {
    var single: UInt? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UInt
}

/**
 * Returns the single element
 * matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun ULongArray.single(predicate: (ULong) -> Boolean): ULong {
    var single: ULong? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as ULong
}

/**
 * Returns the single element
 * matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun UByteArray.single(predicate: (UByte) -> Boolean): UByte {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UByte
}

/**
 * Returns the single element
 * matching the given [predicate], or throws exception if there is no or more than one matching element.
 */
public inline fun UShortArray.single(predicate: (UShort) -> Boolean): UShort {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) throw IllegalArgumentException("Array contains more than one matching element.")
            single = element
            found = true
        }
    }
    if (!found) throw NoSuchElementException("Array contains no element matching the predicate.")
    return single as UShort
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun UIntArray.singleOrNull(): UInt? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun ULongArray.singleOrNull(): ULong? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun UByteArray.singleOrNull(): UByte? {
    return if (size == 1) this[0] else null
}

/**
 * Returns single element, or `null` if the array is empty or has more than one element.
 */
public fun UShortArray.singleOrNull(): UShort? {
    return if (size == 1) this[0] else null
}
false for (element in this) { if (predicate(element)) { if (found) return null
    } single = element
    found = true
    } } if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or 'null' if element was not found or more than one element was found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UByteArray.singleOrNull(predicate: (UByte) -> Boolean): UByte? {
    var single: UByte? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns the single element matching the given [predicate], or 'null' if element was not found or more than one element was found.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun UShortArray.singleOrNull(predicate: (UShort) -> Boolean): UShort? {
    var single: UShort? = null
    var found = false
    for (element in this) {
        if (predicate(element)) {
            if (found) return null
            single = element
            found = true
        }
    }
    if (!found) return null
    return single
}

/**
 * Returns a list containing all elements except first [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.drop(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return takeLast((size - n).coerceAtLeast(0))\n}

/**
 * Returns a list containing all elements except last [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.dropLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n}

/**
 * Returns a list containing all elements except last [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.dropLast(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n}

/**
 * Returns a list containing all elements except first [n] elements.\n * @throws IllegalArgumentException if [n] is negative.\n * \n * @sample samples.collections.Collections.Transformations.drop\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.dropLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    return take((size - n).coerceAtLeast(0))\n}
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.dropLast(n: Int):
    List<Short> {
        require(n >= 0) { "Requested element count $n is less than zero." }
        return take((size - n).coerceAtLeast(0))
    }

/**
 * Returns a list containing all elements except last elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.dropLastWhile(predicate: (UByte) -> Boolean):
    List<UByte> {
        for (index in lastIndex downTo 0)
            if (!predicate(this[index])) {
                return take(index + 1)
            }
        return emptyList()
    }

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.dropWhile(predicate: (UShort) -> Boolean):
    List<UShort> {
        var yielding = false
        val list = ArrayList<UShort>()
        for (item in this)
            if (yielding)
                list.add(item)
            else if (!predicate(item)) {
                list.add(item)
                yielding = true
            }
        return list
    }

/**
 * Returns a list containing all elements except first elements that satisfy the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArrayList.dropWhile(predicate: (UByte) -> Boolean):
    List<UByte> {
        var yielding = false
        val list = ArrayList<UByte>()
        for (item in this)
            if (yielding)
                list.add(item)
            else if (!predicate(item)) {
                list.add(item)
                yielding = true
            }
        return list
    }

/**
 * Returns a list containing only elements matching the given [predicate].
 * 
 * @sample samples.collections.Collections.Transformations.drop
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArrayList.dropWhile(predicate: (UShort) -> Boolean):
    List<UShort> {
        var yielding = false
        val list = ArrayList<UShort>()
        for (item in this)
            if (yielding)
                list.add(item)
            else if (!predicate(item)) {
                list.add(item)
                yielding = true
            }
        return list
    }

/*
 samples.collections.Collections.Transformations.drop
*/

/*
 samples.collections.Collections.Transformations.drop
*/

/*
 samples.collections.Collections.Transformations.drop
*/

/*
 samples.collections.Collections.Transformations.drop
*/

/*
 samples.collections.Collections.Transformations.drop
*/
*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UIntArray.filter(predicate: (UInt) -> Boolean): List<UInt> {  
    return filterTo(ArrayList<UInt>(), predicate)\n* Returns a list containing only elements matching the given [predicate].\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun ULongArray.filter(predicate: (ULong) -> Boolean): List<ULong> {  
    return filterTo(ArrayList<ULong>(), predicate)\n* Returns a list containing only elements matching the given [predicate].\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UByteArray.filter(predicate: (UByte) -> Boolean): List<UByte> {  
    return filterTo(ArrayList<UByte>(), predicate)\n* Returns a list containing only elements matching the given [predicate].\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UShortArray.filter(predicate: (UShort) -> Boolean): List<UShort> {  
    return filterTo(ArrayList<UShort>(), predicate)\n* Returns a list containing only elements matching the given [predicate].\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UIntArray.filterIndexed(predicate: (index: Int, UInt) -> Boolean): List<UInt> {  
    return filterIndexedTo(ArrayList<UInt>(), predicate)\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun ULongArray.filterIndexed(predicate: (index: Int, ULong) -> Boolean): List<ULong> {  
    return filterIndexedTo(ArrayList<ULong>(), predicate)\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UByteArray.filterIndexed(predicate: (index: Int, UByte) -> Boolean): List<UByte> {  
    return filterIndexedTo(ArrayList<UByte>(), predicate)\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun UShortArray.filterIndexed(predicate: (index: Int, UShort) -> Boolean): List<UShort> {  
    return filterIndexedTo(ArrayList<UShort>(), predicate)\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun <C : MutableCollection<in UInt>> UIntArray.filterIndexedTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {  
    forEachIndexed { index, element ->  
        if (predicate(index, element)) destination.add(element)\n    }\n  return destination\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun <C : MutableCollection<in ULong>> ULongArray.filterIndexedTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {  
    forEachIndexed { index, element ->  
        if (predicate(index, element)) destination.add(element)\n    }\n  return destination\n* @param [predicate] function that takes the index of an element and the element itself\n* and returns the result of predicate evaluation on the element.\n  }

*\@SinceKotlin("1.3")\@ExperimentalUnsignedTypes\@kotlin.internal.InlineOnly\public inline fun <C : MutableCollection<in UByte>> UByteArray.filterIndexedTo(destination: C, predicate: (index: Int, UByte) ->
Boolean): C {
  forEachIndexed { index, element ->
    if (predicate(index, element))
      destination.add(element)
    return destination
  }
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun <C : MutableCollection<in UShort>> UShortArray.filterIndexedTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
  forEachIndexed { index, element ->
    if (predicate(index, element))
      destination.add(element)
  }
  return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
  for (element in this) if (!predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun UIntArray.filterNotTo(destination: MutableList<in UInt>, predicate: (index: Int, UInt) -> Boolean): MutableList<in UInt> {
  for (element in this) if (!predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun ULongArray.filterNotTo(destination: MutableList<in ULong>, predicate: (index: Int, ULong) -> Boolean): MutableList<in ULong> {
  for (element in this) if (!predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun UByteArray.filterNotTo(destination: MutableList<in UByte>, predicate: (index: Int, UByte) -> Boolean): MutableList<in UByte> {
  for (element in this) if (!predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Returns a list containing all elements not matching the given [predicate].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
  for (element in this) if (!predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun <C : MutableCollection<in UInt>> UIntArray.filterTo(destination: C, predicate: (index: Int, UInt) -> Boolean): C {
  for (element in this) if (predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun <C : MutableCollection<in ULong>> ULongArray.filterTo(destination: C, predicate: (index: Int, ULong) -> Boolean): C {
  for (element in this) if (predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun <C : MutableCollection<in UByteArray>> UByteArray.filterTo(destination: C, predicate: (index: Int, UByte) -> Boolean): C {
  for (element in this) if (predicate(index, element)) destination.add(element)
  return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 *------------------------------------------------------------------*
 * @param [predicate] function that takes the index of an element and the element itself
 * and returns the result of predicate evaluation on the element.
 *------------------------------------------------------------------*
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@InlineOnly
public inline fun <C : MutableCollection<in UShortArray>> UShortArray.filterTo(destination: C, predicate: (index: Int, UShort) -> Boolean): C {
  for (element in this) if (predicate(index, element)) destination.add(element)
  return destination
}
(element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@kotlin.experimental-inline
@kotlin.internal.InlineOnly
fun List<? super UShort>.filterTo(destination: C, predicate: (UShort) -> Boolean): C {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Appends all elements matching the given [predicate] to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UShortArray.filterTo(destination: MutableList<UShort>, predicate: (UShort) -> Boolean): MutableList<UShort> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UByteFilterTo(destination: MutableList<UByte>, predicate: (UByte) -> Boolean): MutableList<UByte> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun ULongFilterTo(destination: MutableList<ULong>, predicate: (ULong) -> Boolean): MutableList<ULong> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UByteArray(filterTo(destination: MutableList<UByte>, predicate: (UByte) -> Boolean): MutableList<UByte> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UShortArray(filterTo(destination: MutableList<UShort>, predicate: (UShort) -> Boolean): MutableList<UShort> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UIntArray(filterTo(destination: MutableList<UInt>, predicate: (UInt) -> Boolean): MutableList<UInt> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun ULongArray(filterTo(destination: MutableList<ULong>, predicate: (ULong) -> Boolean): MutableList<ULong> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UByteArray(filterTo(destination: MutableList<UByte>, predicate: (UByte) -> Boolean): MutableList<UByte> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns a list containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UShortArray(filterTo(destination: MutableList<UShort>, predicate: (UShort) -> Boolean): MutableList<UShort> {
    for (element in this) if (predicate(element)) destination.add(element)
    return destination
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UIntArray.sliceArray(indices: Collection<Int>): UIntArray {
    return UIntArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun ULongArray.sliceArray(indices: Collection<Int>): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UByteArray.sliceArray(indices: Collection<Int>): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
fun UShortArray.sliceArray(indices: Collection<Int>): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}
indices in the specified [indices] range.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.sliceArray(indices: IntRange): ULongArray {
    return ULongArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.sliceArray(indices: IntRange): UByteArray {
    return UByteArray(storage.sliceArray(indices))
}

/**
 * Returns an array containing elements at indices in the specified [indices] range.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.sliceArray(indices: IntRange): UShortArray {
    return UShortArray(storage.sliceArray(indices))
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.take(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UInt>(n)
    for (item in this) {
        list.add(item)
        if (++count == n)
            break
    }
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.take(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.take(n: Int): List<UByte> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UByte>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing first [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.take(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == 1) return listOf(this[0])
    var count = 0
    val list = ArrayList<UShort>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.takeLast(n: Int): List<UInt> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<UInt>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.takeLast(n: Int): List<ULong> {
    require(n >= 0) { "Requested element count $n is less than zero." }
    if (n == 0) return emptyList()
    if (n == size) return toList()
    if (n == 1) return listOf(this[size - 1])
    val list = ArrayList<ULong>(n)
    for (index in size - n until size)
        list.add(this[index])
    return list
}

/**
 * Returns a list containing last [n] elements.
 * @throws IllegalArgumentException if [n] is negative.
 * @sample samples.collections.Collections.Transformations.take
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.takeLast(n: Int): List<UByte>
```kotlin
{\n    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    val size = size\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[size - 1])\n    val list = ArrayList<UByte>(n)\n    for (index in size - n until size)\n        list.add(this[index])\n    return list
}

/**
 * Returns a list containing last \[n\] elements.
 * 
 * @throws IllegalArgumentException if \[n\] is negative.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.takeLast(n: Int): List<UShort> {
    require(n >= 0) { "Requested element count $n is less than zero." }\n    if (n == 0) return emptyList()\n    val size = size\n    if (n > size) return toList()\n    if (n == 1) return listOf(this[size - 1])\n    val list = ArrayList<UShort>(n)\n    for (index in size - n until size)\n        list.add(this[index])\n    return list
}

/**
 * Returns a list containing last elements satisfying the given \[predicate\].\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeLastWhile(predicate: (UByte) -> Boolean): List<UByte> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing last elements satisfying the given \[predicate\].\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeLastWhile(predicate: (UShort) -> Boolean): List<UShort> {
    for (index in lastIndex downTo 0) {
        if (!predicate(this[index])) {
            return drop(index + 1)
        }
    }
    return toList()
}

/**
 * Returns a list containing first elements satisfying the given \[predicate\].\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.takeWhile(predicate: (UByte) -> Boolean): List<UByte> {
    val list = ArrayList<UByte>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

/**
 * Returns a list containing first elements satisfying the given \[predicate\].\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.takeWhile(predicate: (UShort) -> Boolean): List<UShort> {
    val list = ArrayList<UShort>()
    for (item in this) {
        if (!predicate(item))
            break
        list.add(item)
    }
    return list
}

*/
```
Reverses elements in the array in-place.

```java
    public fun UIntArray.reversed(): List<UInt> {
        if (isEmpty()) return emptyList()
        val list = toMutableList()
        list.reverse()
        return list
    }

    public fun ULongArray.reversed(): List<ULong> {
        if (isEmpty()) return emptyList()
        val list = toMutableList()
        list.reverse()
        return list
    }

    public fun UByteArray.reversed(): List<UByte> {
        if (isEmpty()) return emptyList()
        val list = toMutableList()
        list.reverse()
        return list
    }
```

Returns a list with elements in reversed order.

```java
    public fun UIntArray.reversedArray(): UIntArray {
        return UIntArray(storage.reversedArray())
    }

    public fun ULongArray.reversedArray(): ULongArray {
        return ULongArray(storage.reversedArray())
    }

    public fun UByteArray.reversedArray(): UByteArray {
        return UByteArray(storage.reversedArray())
    }
```

Sorts elements in the array in-place descending according to their natural sort order.

```java
    public fun UIntArray.sorted(): List<UInt> {
        return copyOf().apply { sort() } asList()
    }

    public fun ULongArray.sorted(): List<ULong> {
        return copyOf().apply { sort() } asList()
    }

    public fun UByteArray.sorted(): List<UByte> {
        return copyOf().apply { sort() } asList()
    }
```
elements sorted according to their natural sort order.

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UShortArray.sorted(): List<UShort> {\n    return copyOf().apply { sort() }.asList()\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UIntArray.sortedArray(): UIntArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sort() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun ULongArray.sortedArray(): ULongArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sort() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UByteArray.sortedArray(): UByteArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sort() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UShortArray.sortedArray(): UShortArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sort() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UByteArray.sortedArrayDescending(): UByteArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sortDescending() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UShortArray.sortedArrayDescending(): UShortArray {\n    if (isEmpty()) return this\n    return this.copyOf().apply { sortDescending() }\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UByteArray.asByteArray(): ByteArray {\n    return storage\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UIntArray.asIntArray(): IntArray {\n    return storage\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UIntArray.asList(): List<UInt> {\n    return copyOf().apply { sortDescending() }.reversed()\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UShortArray.asList(): List<UInt> {\n    return copyOf().apply { sortDescending() }.reversed()\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public fun UByteArray.asByteArrayOf(): ByteArray {\n    return storage\n}

*\@SinceKotlin("1.3")\n*\@ExperimentalUnsignedTypes\n*public expect fun UIntArray.asList(): List<UInt>
List<UInt>

/**
 * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun ULongArray.asList(): List<ULong>

/**
 * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UByteArray.asList(): List<UByte>

/**
 * Returns a [List] that wraps the original array.

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public expect fun UShortArray.asList(): List<UShort>

/**
 * Returns an array of type [LongArray], which is a view of this array where each element is
 * a signed reinterpretation

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.asLongArray(): LongArray {
    return storage
}

/**
 * Returns an array of type [ShortArray], which is a view of this array where each element is an
 * unsigned reinterpretation

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.asShortArray(): ShortArray {
    return storage
}

/**
 * Returns an array of type [UByteArray], which is a view of this array where each element is an
 * unsigned reinterpretation

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.asUByteArray(): UByteArray {
    return UByteArray(this)
}

/**
 * Returns an array of type [UIntArray], which is a view of this array where each element is an
 * unsigned reinterpretation

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun IntArray.asUIntArray(): UIntArray {
    return UIntArray(this)
}

/**
 * Returns an array of type [ULongArray], which is a view of this array where each element is an
 * unsigned reinterpretation

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun LongArray.asULongArray(): ULongArray {
    return storage
}

/**
 * Returns a hash code based on the contents of this array as if it is

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.contentHashCode(): Int {
    return storage.contentHashCode()
}

/**
 * Returns a hash code based on the contents of this array as if it is

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.contentHashCode(): Int {
    return storage.contentHashCode()
}
**/**
* Returns a hash code based on the contents of this array as if it is 
[List].

public fun UShortArray.contentHashCode():

    return storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified 
array as if it is [List].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.contentToString(): String {

    return joinToString(", ", "[", "]")
}

/**
 * Returns a hash code based on the contents of this array as if it is 
[List].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.contentHashCode():

    return storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified 
array as if it is [List].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.contentToString(): String {

    return joinToString(", ", "[", "]")
}

/**
 * Returns a string representation of the contents of the specified 
array as if it is [List].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.contentHashCode():

    return storage.contentHashCode()
}

/**
 * Returns a string representation of the contents of the specified 
array as if it is [List].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.contentToString(): String {

    return joinToString(", ", "[", "]")
}
public inline fun UByteArray.copyInto(destination: UByteArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UByteArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Copies this array or its subrange into the [destination] array and returns that array.
 * It's allowed to pass the same array in the [destination] and even specify the subrange so that it overlaps with the
 * destination range.
 * @param destination the array to copy to.
 * @param startIndex the position in the [destination] array to copy to, 0 by default.
 * @param endIndex the end (exclusive) of the subrange to copy, 0 by default.
 * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this array indices or when `startIndex > endIndex`.
 * @return the [destination] array.
 */
public inline fun UShortArray.copyInto(destination: UShortArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = size): UShortArray {
    storage.copyInto(destination.storage, destinationOffset, startIndex, endIndex)
    return destination
}

/**
 * Returns new array which is a copy of the original array.
 * @sample samples.collections.Arrays.CopyOfOperations.copyOf
 */
public inline fun UIntArray.copyOf(): UIntArray {
    return UIntArray(storage.copyOf())
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param newSize the size of the new array.
 * @return the new array.
 */
public inline fun ULongArray.copyOf(newSize: Int): ULongArray {
    return ULongArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param newSize the size of the new array.
 * @return the new array.
 */
public inline fun UByteArray.copyOf(newSize: Int): UByteArray {
    return UByteArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param newSize the size of the new array.
 * @return the new array.
 */
public inline fun UShortArray.copyOf(newSize: Int): UShortArray {
    return UShortArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param newSize the size of the new array.
 * @return the new array.
 */
public inline fun UByteArray.copyOf(newSize: Int): UByteArray {
    return UByteArray(storage.copyOf(newSize))
}

/**
 * Returns new array which is a copy of the original array, resized to the given [newSize].
 * The copy is either truncated or padded at the end with zero values if necessary.
 * @param newSize the size of the new array.
 * @return the new array.
 */
public inline fun UShortArray.copyOf(newSize: Int): UShortArray {
    return UShortArray(storage.copyOf(newSize))
}
original array, the copy array is truncated to the [newSize].

* @SinceKotlin("1.3")
* @ExperimentalUnsignedTypes

public inline fun UShortArray.copyOf(newSize: Int): UShortArray {
    return UShortArray(storage.copyOf(newSize))
}

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 *
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.copyOfRange(fromIndex: Int, toIndex: Int): UIntArray {
    return UIntArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Returns a new array which is a copy of the specified range of the original array.
 * @param fromIndex the start of the range (inclusive), must be in `0..array.size`
 * @param toIndex the end of the range (exclusive), must be in `fromIndex..array.size`
 *
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.copyOfRange(fromIndex: Int, toIndex: Int): ULongArray {
    return ULongArray(storage.copyOfRange(fromIndex, toIndex))
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.fill(element: UInt, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toInt(), fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.fill(element: ULong, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toLong(), fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.fill(element: UByte, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toByte(), fromIndex, toIndex)
}

/**
 * Fills this array or its subrange with the specified [element] value.
 * @param fromIndex the start of the range (inclusive), 0 by default.
 * @param toIndex the end of the range (exclusive), size of this array by default.
 * @throws IndexOutOfBoundsException if [fromIndex] is less than zero or [toIndex] is greater than the size of this array.
 * @throws IllegalArgumentException if [fromIndex] is greater than [toIndex].

 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.fill(element: UShort, fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element, fromIndex, toIndex)
}
fromIndex: Int = 0, toIndex: Int = size): Unit {
    storage.fill(element.toShort(), fromIndex, toIndex)
}

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UIntArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val ULongArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UByteArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the range of valid indices for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.indices: IntRange
    get() = storage.indices

/**
 * Returns the last valid index for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UIntArray.lastIndex: Int
    get() = storage.lastIndex

/**
 * Returns the last valid index for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val ULongArray.lastIndex: Int
    get() = storage.lastIndex

/**
 * Returns the last valid index for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UByteArray.lastIndex: Int
    get() = storage.lastIndex

/**
 * Returns the last valid index for the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline val UShortArray.lastIndex: Int
    get() = storage.lastIndex

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UIntArray.plus(element: UInt): UIntArray {
    return UIntArray(storage +
        element.toInt())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun ULongArray.plus(element: ULong): ULongArray {
    return ULongArray(storage +
        element.toLong())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UByteArray.plus(element: UByte): UByteArray {
    return UByteArray(storage +
        element.toByte())
}

/**
 * Returns an array containing all elements of the original array and then the given
 * [element].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
operator fun UShortArray.plus(element: UShort): UShortArray {
    return UShortArray(storage +
        element.toShort())
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UIntArray.plus(elements: Collection<UInt>): UIntArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toInt()
    return UIntArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun ULongArray.plus(elements: Collection<ULong>): ULongArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toLong()
    return ULongArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UByteArray.plus(elements: Collection<UByte>): UByteArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toByte()
    return UByteArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] collection.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public operator fun UShortArray.plus(elements: Collection<UShort>): UShortArray {
    var index = size
    val result = storage.copyOf(size + elements.size)
    for (element in elements) result[index++] = element.toShort()
    return UShortArray(result)
}

/**
 * Returns an array containing all elements of the original array and then all elements
 * of the given [elements] array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun

---

Open Source Used in IOx-FogDirector-AC18 1.17.0 2253

---
UIntArray.plus(elements: UIntArray): UIntArray {
    return UIntArray(storage + elements.storage)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongArray.plus(elements: ULongArray): ULongArray {
    return ULongArray(storage + elements.storage)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UByteArray.plus(elements: UByteArray): UByteArray {
    return UByteArray(storage + elements.storage)
}

/**
 * Returns an array containing all elements of the original array and then all elements of the given [elements] array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UShortArray.plus(elements: UShortArray): UShortArray {
    return UShortArray(storage + elements.storage)
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Sorts the array in-place.
 * 
 * @sample samples.collections.Arrays.Sorting.sortArray
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.sort(): Unit {
    if (size > 1) sortArray(this)
}

/**
 * Returns an array of type [ByteArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.toByteArray(): ByteArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [IntArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.toIntArray(): IntArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [LongArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.toLongArray(): LongArray {
    return storage.copyOf()
}

/**
 * Returns an array of type [ShortArray], which is a copy of this array where each element is a signed reinterpretation
 * of the corresponding element of this array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.toShortArray(): ShortArray {
    return storage.copyOf()
}

/**
 * Returns a typed object array containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UIntArray.toTypedArray(): Array<UInt> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns a typed object array containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun ULongArray.toTypedArray(): Array<ULong> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns a typed object array containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UByteArray.toTypedArray(): Array<UByte> {
    return Array(size) { index -> this[index] }
}

/**
 * Returns a typed object array containing all of the elements of this generic array.
 */
@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public fun UShortArray.toTypedArray(): Array<UShort> {
    return Array(size) { index -> this[index] }
}
Array<out UByte>.toUByteArray(): UByteArray {
    return UByteArray(size) { index -> this[index] }
}

* Returns an array of type [UByteArray], which is a copy of this array where each element is an unsigned reinterpretation

* Returns an array of the corresponding element of this array.

*/

@SinceKotlin("1.3")
@IEEE754
@kotlin.internal.InlineOnly
@public fun ByteArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
}

/**
 * Returns an array of type [Semaphore], which is a copy of this array where each element is an unsigned reinterpretation
 * of the corresponding element of this array.

*/

@SinceKotlin("1.3")
@IEEE754
@kotlin.internal.InlineOnly
@public fun IntArray.toUByteArray(): UByteArray {
    return UByteArray(this.copyOf())
}

/**
 * Returns a single list of all elements yielded from results of [transform] function being invoked on each element of original array.

*/
@SinceKotlin("1.3")
@IEEE754
@kotlin.internal.InlineOnly
@public fun <R> UByteArray.flatMap(transform: (UByte) -> Iterable<R>): List<R> {
    return flatMapTo(ArrayList<R>(), transform)
}

*/

Open Source Used in IOx-FogDirector-AC18 1.17.0 2255

C : MutexCollection<br>
for
/**
 * Appends all elements yielded from results of [transform] function being invoked on each element of original array, to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>> ULongArray.flatMapTo(destination: C, transform: (ULong) -> Iterable<R>): C {
    for (element in this) {
        val list = transform(element)
        destination.addAll(list)
    }
    return destination
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> {
    return groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UByteArray.groupBy(keySelector: (UByte) -> K): Map<K, List<UByte>> {
    return groupByTo(LinkedHashMap<K, MutableList<UByte>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}

/**
 * Groups elements of the original array by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K>
    UShortArray.groupBy(keySelector: (UShort) -> K): Map<K, List<UShort>> {
    return groupByTo(LinkedHashMap<K, MutableList<UShort>>(), keySelector)
}

/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array
 * by the key returned by the given [keySelector] function
 * applied to each element and returns a map where each group key is associated with a list of corresponding values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V>
UIntArray.groupBy(keySelector: (UInt) -> K, valueTransform: (UInt) -> V): Map<K, List<V>>

{\n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> ULongArray.groupBy(keySelector: (ULong) -> K, valueTransform: (ULong) -> V): Map<K, List<V>>

{\n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> UByteArray.groupBy(keySelector: (UByte) -> K, valueTransform: (UByte) -> V): Map<K, List<V>>

{\n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n
/**
 * Groups values returned by the [valueTransform] function applied to each element of the original array by the key returned by the given [keySelector] function applied to the element and returns a map where each group key is associated with a list of corresponding values.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, V> UShortArray.groupBy(keySelector: (UShort) -> K, valueTransform: (UShort) -> V): Map<K, List<V>>

{\n    return groupByTo(LinkedHashMap<K, MutableList<V>>(), keySelector, valueTransform)\n}\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<UInt>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K): M

{\n    for (element in this)\n        val key = keySelector(element)\n        val list = destination.getOrElse(key) \n        list.add(element)\n    return destination\n}\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<ULong>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K): M

{\n    for (element in this)\n        val key = keySelector(element)\n        val list = destination.getOrElse(key) \n        list.add(element)\n    return destination\n}\n
/**
 * Groups elements of the original array by the key returned by the given [keySelector] function applied to each element and puts to the [destination] map each group key associated with a list of corresponding elements.\n * The returned map preserves the entry iteration order of the keys produced from the original array.\n */

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<UByte>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K): M

{\n    for (element in this)\n        val key = keySelector(element)\n        val list = destination.getOrElse(key) \n        list.add(element)\n    return destination\n}\n
Open Source Used In IOx-FogDirector-AC18 1.17.0 2257
[destination] map each group key associated with a list of corresponding elements.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<in V>>> UShortArray.groupByTo(destination: M, keySelector: (UShort) -> K):
M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<UShort>()
        }
        list.add(element)
    }
    return destination
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<in V>>> UIntArray.groupByTo(destination: M, keySelector: (UInt) -> K,
valueTransform: (UInt) -> V):
M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<in V>>> ULongArray.groupByTo(destination: M, keySelector: (ULong) -> K,
valueTransform: (ULong) -> V):
M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <K, M : MutableMap<in K, MutableList<in V>>> UByteArray.groupByTo(destination: M, keySelector: (UByte) -> K,
valueTransform: (UByte) -> V):
M {
    for (element in this) {
        val key = keySelector(element)
        val list = destination.getOrPut(key) {
            ArrayList<V>()
        }
        list.add(valueTransform(element))
    }
    return destination
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.map(transform: (UInt) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.map(transform: (ULong) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```

/* Groups values returned by the [valueTransform] function applied to each element of the original array */

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}
```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.map(transform: (UByte) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UShortArray.map(transform: (UShort) -> R): List<R> {
    return mapTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.mapIndexed(transform: (index: Int, UInt) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.mapIndexed(transform: (index: Int, ULong) -> R): List<R> {
    return mapIndexedTo(ArrayList<R>(size), transform)
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UIntArray.mapIndexedTo(destination: C, transform: (index: Int, UInt) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
ULongArray.mapIndexedTo(destination: C, transform: (index: Int, ULong) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UByteArray.mapIndexedTo(destination: C, transform: (index: Int, UByte) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Returns a list containing the results of applying the given [transform] function\n * to each element and its index in the original array.\n * @param [transform] function that takes the index of an element and the element itself\n * and returns the result of the transform applied to the element.\n */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C : MutableCollection<in R>>
UShortArray.mapIndexedTo(destination: C, transform: (index: Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}
```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :MutableCollection<in R>> UShortArray.mapIndexedTo(destination: C, transform: (Int, UShort) -> R): C {
    var index = 0
    for (item in this)
        destination.add(transform(index++, item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :MutableCollection<in R>> UIntArray.mapTo(destination: C, transform: (UInt) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :MutableCollection<in R>> ULongArray.mapTo(destination: C, transform: (ULong) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :MutableCollection<in R>> UByteArray.mapTo(destination: C, transform: (UByte) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Applies the given [transform] function to each element of the original array
 * and appends the results to the given [destination].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, C :MutableCollection<in R>> UShortArray.mapTo(destination: C, transform: (UShort) -> R): C {
    for (item in this)
        destination.add(transform(item))
    return destination
}

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntArray.withIndex():
    Iterable<IndexedValue<UInt>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongArray.withIndex():
    Iterable<IndexedValue<ULong>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByteArray.withIndex():
    Iterable<IndexedValue<UByte>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns a lazy [Iterable] that wraps each element of the original array
 * into an [IndexedValue] containing the index of that element and the element
 * itself. 
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.withIndex():
    Iterable<IndexedValue<UShort>> {
        return IndexingIterable { iterator() }
    }

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
    UIntArray.all(predicate: (UInt) -> Boolean): Boolean {
        for (element in this) if (!predicate(element)) return false
        return true
    }

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
    ULongArray.all(predicate: (ULong) -> Boolean): Boolean {
        for (element in this) if (!predicate(element)) return false
        return true
    }

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
    UByteArray.all(predicate: (UByte) -> Boolean): Boolean {
        for (element in this) if (!predicate(element)) return false
        return true
    }

/**
 * Returns `true` if all elements match the given [predicate].
 * 
 * @sample samples.collections.Collections.Aggregates.all
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
    UShortArray.all(predicate: (UShort) -> Boolean): Boolean {
        for (element in this) if (!predicate(element)) return false
        return true
    }
```

---

Open Source Used In IOx-FogDirector-AC18 1.1.7.0 2260
samples.collections.Collections.Aggregates.any

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if array has at least one element.
 * @sample samples.collections.Collections.Aggregates.any
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.any(): Boolean {
    return storage.any()
}

/**
 * Returns `true` if at least one element matches the given [predicate]. Returns `true` if at least one element matches the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.any(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns `true` if at least one element matches the given [predicate]. Returns `true` if at least one element matches the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.any(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return true
    return false
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UIntArray.count(predicate: (UInt) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
ULongArray.count(predicate: (ULong) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UByteArray.count(predicate: (UByte) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Returns the number of elements matching the given [predicate].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun
UShortArray.count(predicate: (UShort) -> Boolean): Int {
    var count = 0
    for (element in this) if (predicate(element)) ++count
    return count
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UIntArray.fold(initial: R, operation: (acc: R, UInt) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
ULongArray.fold(initial: R, operation: (acc: R, ULong) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from left to right to current accumulator value and each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R>
    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)
    return accumulator
}
starting with [initial] value and applying [operation] from left to right to current accumulator value and each
element.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> UByteArray.fold(initial: R, operation: (acc: R, UByte) -> R): R {

    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline

    var accumulator = initial
    for (element in this) accumulator = operation(accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> UIntArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UInt) -> R): R {

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> ULongArray.foldIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): R {

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> UByteArray.foldIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline

    var index = 0
    var accumulator = initial
    for (element in this) accumulator = operation(index++, accumulator, element)

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> UIntArray.foldRight(initial: R, operation: (UInt, acc: R) -> R): R {

    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> ULongArray.foldRight(initial: R, operation: (ULong, acc: R) -> R): R {

    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline
fun <R> UByteArray.foldRight(initial: R, operation: (index: Int, acc: R, UByte) -> R): R {

    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }

    return accumulator
}
```

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline

    var index = lastIndex
    var accumulator = initial
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }

    return accumulator
}
```
UByteArray.foldRight(initial: R, operation: (UByte, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(get(index--), accumulator)
  }
  return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right
 * to left to each element and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UShortArray.foldRight(initial: R, operation: (UShort, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(get(index--), accumulator)
  }
  return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right
 * to left to each element with its index in the original array and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UIntArray.foldRightIndexed(initial: R, operation: (Int, UInt, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(index, get(index), accumulator)
    --index
  }
  return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right
 * to left to each element with its index in the original array and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> ULongArray.foldRightIndexed(initial: R, operation: (Int, ULong, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(index, get(index), accumulator)
    --index
  }
  return accumulator
}

/**
 * Accumulates value starting with [initial] value and applying [operation] from right
 * to left to each element with its index in the original array and current accumulator value.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R> UByteArray.foldRightIndexed(initial: R, operation: (Int, UByte, acc: R) -> R): R {
  var index = lastIndex
  var accumulator = initial
  while (index >= 0) {
    accumulator = operation(index, get(index), accumulator)
    --index
  }
  return accumulator
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <E> UIntArray.forEach(action: (UInt) -> Unit): Unit {
  for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <E> ULongArray.forEach(action: (ULong) -> Unit): Unit {
  for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <E> UByteArray.forEach(action: (UByte) -> Unit): Unit {
  for (element in this) action(element)
}

/**
 * Performs the given [action] on each element.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <E> UShortArray.forEach(action: (UShort) -> Unit): Unit {
  for (element in this) action(element)
}
function that takes the index of an element and the element itself\n* and performs the desired action on the element.\n*
function that takes the index of an element and the element itself\n* and performs the desired action on the element.\n*
function that takes the index of an element and the element itself\n* and performs the desired action on the element.\n*
function that takes the index of an element and the element itself\n* and performs the desired action on the element.\n*
```kotlin
maxElem = this[0]
val lastIndex = this.lastIndex
for (i in 1..lastIndex) {
    val e = this[i]
    val v = selector(e)
    if (maxValue < v)
        maxElem = e
        maxValue = v
}

val = this[i]
val v = selector(e)
if (maxValue < v)
    maxElem = e
    maxValue = v

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public inline fun <R : Comparable<R>> UShortArray.maxBy(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var maxElem = this[0]
    val lastIndex = this.lastIndex
    if (lastIndex == 0) return maxElem
    var maxValue = selector(maxElem)
    for (i in 1..lastIndex) {
        val e = this[i]
        val v = selector(e)
        if (maxValue < v)
            maxElem = e
            maxValue = v
    }
    return maxElem
}

/**
 * Returns the first element having the largest value according to the provided [comparator] or ‘null’ if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
    if (isEmpty()) return null
    var max = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (comparator.compare(max, e) < 0) max = e
    }
    return max
}

/**
 * Returns the smallest element or ‘null’ if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShortArray.min(): UShort? {
    if (isEmpty()) return null
    var min = this[0]
    for (i in 1..lastIndex) {
        val e = this[i]
        if (min > e) min = e
    }
    return min
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2265
Comparable<R>> ULongArray.minBy(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var minElem = this[0]
    var lastIndexOf = this.lastIndex
    if (lastIndexOf == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1...lastIndexOf) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Comparable<R>> ULongArray.minBy(selector: (ULong) -> R): ULong? {
    if (isEmpty()) return null
    var minElem = this[0]
    var lastIndexOf = this.lastIndex
    if (lastIndexOf == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1...lastIndexOf) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Comparable<R>> UByteArray.minBy(selector: (UByte) -> R): UByte? {
    if (isEmpty()) return null
    var minElem = this[0]
    var lastIndexOf = this.lastIndex
    if (lastIndexOf == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1...lastIndexOf) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}

/**
 * Returns the first element yielding the smallest value of the given function or `null` if there are no elements.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun Comparable<R>> UShortArray.minBy(selector: (UShort) -> R): UShort? {
    if (isEmpty()) return null
    var minElem = this[0]
    var lastIndexOf = this.lastIndex
    if (lastIndexOf == 0) return minElem
    var minValue = selector(minElem)
    for (i in 1...lastIndexOf) {
        val e = this[i]
        val v = selector(e)
        if (minValue > v) {
            minElem = e
            minValue = v
        }
    }
    return minElem
}
`UIntArray.none(predicate: (UInt) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given `predicate`. from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.none(predicate: (ULong) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given `predicate`. from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.none(predicate: (UByte) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Returns `true` if no elements match the given `predicate`. from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.noneWithPredicate
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.none(predicate: (UShort) -> Boolean): Boolean {
    for (element in this) if (predicate(element)) return false
    return true
}

/**
 * Accumulates value starting with the first element and applying `operation` from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduce(operation: (acc: UInt, UInt) -> UInt): UInt {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying `operation` from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduce(operation: (acc: ULong, ULong) -> ULong): ULong {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying `operation` from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduce(operation: (acc: UByte, UByte) -> UByte): UByte {
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}

/**
 * Accumulates value starting with the first element and applying `operation` from left to right to current accumulator value and each element.
 * 
 * @sample samples.collections.Collections.Aggregates.reduce
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty())
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = this[0]
    for (index in 1..lastIndex) {
        accumulator = operation(accumulator, this[index])
    }
    return accumulator
}
function that takes the index of an element, current accumulator value\n* and the element itself and calculates the
next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce
\n*An@SinceKotlin("1.3")\n*An@ExperimentalUnsignedTypes\n*An@kotlin.internal.InlineOnly\n\npublic inline fun
UByteArray.reduceIndexed(operation: (index: Int, acc: UByte, UByte) -> UByte): UByte {\n    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(index, accumulator, this[index])\n    }\n    return accumulator\n}\n
/**
 * Accumulates value starting with the first element and applying [operation] from left to
right\n* to current accumulator value and each element with its index in the original array.\n* @param [operation] function that takes the index of an element, current accumulator value\n* and the element itself and calculates the
next accumulator value.\n* @sample samples.collections.Collections.Aggregates.reduce
\n*An@SinceKotlin("1.3")\n*An@ExperimentalUnsignedTypes\n*An@kotlin.internal.InlineOnly\n\npublic inline fun
UShortArray.reduceIndexed(operation: (index: Int, acc: UShort, UShort) -> UShort): UShort {\n    if (isEmpty()) throw UnsupportedOperationException("Empty array can't be reduced.")\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(index, accumulator, this[index])\n    }\n    return accumulator\n}\n
/**
 * Accumulates value starting with the first element and applying [operation] from left to
right to current accumulator value and each element. Returns null if the array is empty.\n* @sample samples.collections.Collections.Aggregates.reduceOrNull
\n*An@SinceKotlin("1.3")\n*An@ExperimentalUnsignedTypes\n*An@kotlin.internal.InlineOnly\n\npublic inline fun
UIntArray.reduceOrNull(operation: (acc: UInt, UInt) -> UInt): UInt? {\n    if (isEmpty()) return null\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n
/**
 * Accumulates value starting with the last element and applying [operation] from right to left to current accumulator value and each element. Returns null if the array is empty.\n* @sample samples.collections.Collections.Aggregates.reduceOrNull
\n*An@SinceKotlin("1.3")\n*An@ExperimentalUnsignedTypes\n*An@kotlin.internal.InlineOnly\n\npublic inline fun
UByteArray.reduceOrNull(operation: (acc: UByte, UByte) -> UByte): UByte? {\n    if (isEmpty()) return null\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n
/**
 * Accumulates value starting with the last element and applying [operation] from right to left to current accumulator value and each element. Returns null if the array is empty.\n* @sample samples.collections.Collections.Aggregates.reduceOrNull
\n*An@SinceKotlin("1.3")\n*An@ExperimentalUnsignedTypes\n*An@kotlin.internal.InlineOnly\n\npublic inline fun
UShortArray.reduceOrNull(operation: (acc: UShort, UShort) -> UShort): UShort? {\n    if (isEmpty()) return null\n    var accumulator = this[0]\n    for (index in 1..lastIndex) {\n        accumulator = operation(accumulator, this[index])\n    }\n    return accumulator\n}\n
*/

Open Source Used In IOx-FogDirector-AC18 1.17.0 2268
samples.collections.Collections.Aggregates.reduceRight

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntArray.reduceRight(operation: (UInt, acc: UInt) -> UInt): UInt {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
to each element and current accumulator value.

 * @param [operation]
 * function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongArray.reduceRightIndexed(operation: (index: Int, ULong, acc: ULong) -> ULong): ULong {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(index, get(index), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
to each element with its index in the original array and current accumulator value.

 * @param [operation]
 * function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UByteArray.reduceRight(operation: (UByte, acc: UByte) -> UByte): UByte {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}

/**
 * Accumulates value starting with last element and applying [operation] from right to left
to each element with its index in the original array and current accumulator value.

 * @param [operation]
 * function that takes the index of an element, the element itself
 * and current accumulator value, and calculates the next accumulator value.
 * 
 * @sample
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UShortArray.reduceRight(operation: (UShort, acc: UShort) -> UShort): UShort {
    var index = lastIndex
    if (index < 0)
        throw UnsupportedOperationException("Empty array can't be reduced.")
    var accumulator = get(index--)
    while (index >= 0) {
        accumulator = operation(get(index--), accumulator)
    }
    return accumulator
}
UByteArray.reduceRightIndexed(operation: (index: Int, UByte, acc: UByte) -> UByte): UByte

```kotlin
var index = lastIndex
if (index < 0) throw UnsupportedOperationException("Empty array can't be reduced.")
var accumulator = get(index--) while (index >= 0) {
    accumulator = operation(index, get(index), accumulator)
    --index
} return accumulator
```

Accumulates value starting with last element and applying [operation] from right to left. If each element with its index in the original array and current accumulator value.* * @param [operation] function that takes the index of an element, the element itself* and current accumulator value, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.reduceRight

```
@SinceKotlin("1.3")
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.* * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

```
@SinceKotlin("1.3")
public inline fun UIntArray.reduceRightOrNull(operation: (UInt, acc: UInt) -> UInt): UInt?
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.* * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

```
@SinceKotlin("1.3")
public inline fun ULongArray.reduceRightOrNull(operation: (ULong, acc: ULong) -> ULong): ULong?
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.* * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

```
@SinceKotlin("1.3")
public inline fun UByteArray.reduceRightOrNull(operation: (UByte, acc: UByte) -> UByte): UByte?
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.* * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

```
@SinceKotlin("1.3")
public inline fun UShortArray.reduceRightOrNull(operation: (UShort, acc: UShort) -> UShort): UShort?
```

Accumulates value starting with last element and applying [operation] from right to left to each element and current accumulator value. Returns null if the array is empty.* * @sample samples.collections.Collections.Aggregates.reduceRightOrNull

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun UIntArray.scan(initial: UInt, operation: (acc: UInt, UInt) -> UInt): List<UInt>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun ULongArray.scan(initial: ULong, operation: (acc: ULong, ULong) -> ULong): List<ULong>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun UByteArray.scan(initial: UByte, operation: (acc: UByte, UByte) -> UByte): List<UByte>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun UIntArray.scan(initial: UInt, operation: (acc: UInt, UInt) -> UInt): List<UInt>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun ULongArray.scan(initial: ULong, operation: (acc: ULong, ULong) -> ULong): List<ULong>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
public inline fun UByteArray.scan(initial: UByte, operation: (acc: UByte, UByte) -> UByte): List<UByte>
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan

```
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@ExperimentalStdlibApi
```

Returns a list containing successive accumulation values generated by applying [operation] from left to right.* * Note that `acc` value passed to [operation] function should not be mutated.* * otherwise it would affect the previous value in resulting list.* * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.* * @sample samples.collections.Collections.Aggregates.scan
for (element in this) {
    accumulator = operation(accumulator, element)
    result.add(accumulator)
}
return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun ArrayList<T>.scan(initial: T, operation: (acc: T, T) -> T): List<T> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<T>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun <R> ULongArray.scan(initial: R, operation: (acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun <R> UByteArray.scan(initial: R, operation: (acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun <R> UCharArray.scan(initial: R, operation: (acc: R, UChar) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (element in this) {
        accumulator = operation(accumulator, element)
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying
 * [operation] from left to right to each element and current accumulator value that
 * starts with [initial] value.
 * 
 * Note that `acc` value passed to [operation] function should not be mutated;
 * otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an
 * element, and calculates the next accumulator value.
 * @sample samples.collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal:inline
public inline fun <R> UIntArray.scan(initial: R, operation: (acc: R, UInt) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
public inline fun <R> ULongArray.scanIndexed(initial: R, operation: (index: Int, acc: R, ULong) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element, its index in the original array and current accumulator value that starts with [initial] value.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scan
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y
public inline fun <R> UByteArray.scanIndexed(initial: R, operation: (index: Int, acc: R, UByte) -> R): List<R> {
    if (isEmpty()) return listOf(initial)
    val result = ArrayList<R>(size + 1).apply { add(initial) }
    var accumulator = initial
    for (index in indices) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y
public inline fun UIntArray.scanReduce(operation: (acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply { add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
 * Returns a list containing successive accumulation values generated by applying [operation] from left to right
 * to each element and current accumulator value that starts with the first element of this array.
 * Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
 * @param [operation] function that takes current accumulator value and an element, and calculates the next accumulator value.
 * @sample samples.collections.Collections.Aggregates.scanReduce
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y
public inline fun ULongArray.scanReduce(operation: (acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}
public inline fun UByteArray.scanReduce(operation: (acc: UByte, UByte) -> UByte): List<UByte> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UByte>(size).apply {
        add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

public fun UIntArray.scanReduceIndexed(operation: (index: Int, acc: UInt, UInt) -> UInt): List<UInt> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UInt>(size).apply {
        add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

public fun ULongArray.scanReduceIndexed(operation: (index: Int, acc: ULong, ULong) -> ULong): List<ULong> {
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<ULong>(size).apply {
        add(accumulator) }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list. * @param [operation] function that takes the index of an element, current accumulator value and the element itself, and calculates the next accumulator value.

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, its index in the original array and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.

Returns a list containing successive accumulation values generated by applying [operation] from left to right to each element, and current accumulator value that starts with the first element of this array. Note that `acc` value passed to [operation] function should not be mutated; otherwise it would affect the previous value in resulting list.
operation(index, accumulator, this[index])
result.add(accumulator)
} return result

/**
* Returns a list containing successive accumulation values generated by applying \[operation\] from left to right to
* each element, its index in the original array and current accumulator value that starts with the first element of this
* array. Note that `acc` value passed to \[operation\] function should not be mutated; otherwise it would affect the
* previous value in resulting list.
* @param \[operation\] function that takes the index of an element, current accumulator value `acc` and the element itself,
* and calculates the next accumulator value.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    if (isEmpty()) return emptyList()
    var accumulator = this[0]
    val result = ArrayList<UShort>(size).apply {
        add(accumulator)
    }
    for (index in 1 until size) {
        accumulator = operation(index, accumulator, this[index])
        result.add(accumulator)
    }
    return result
}

/**
* Returns the sum of all values produced by \[selector\] function applied to each element in the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntArray.sumBy(selector: (UInt) -> UInt): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
* Returns the sum of all values produced by \[selector\] function applied to each element in the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongArray.sumByDouble(selector: (ULong) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
* Returns the sum of all values produced by \[selector\] function applied to each element in the array.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UByteArray.sumByDouble(selector: (UByte) -> Double): Double {
    var sum: Double = 0.0
    for (element in this) {
        sum += selector(element)
    }
    return sum
}

/**
* Returns a list of pairs built from the elements of `this` array and the \[other\] array with the same index. The
* returned list has length of the shortest collection.
*/
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public infix fun <R> UIntArray.zip(other: Array<out R>): List<Pair<UInt, R>> {
    return zip(other) { t1, t2 -> t1 to t2 }
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2274
the elements of `this` array and the [other] array with the same index. * The returned list has length of the shortest collection. 

Some extracted code from the document:

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> ULongArray.zip(other: Array<R>): List<Pair<ULong, R>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} // Returns a list of pairs built from the elements of `this` array and the [other] array with the same index. * The returned list has length of the shortest collection. 

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> UByteArray.zip(other: Array<R>): List<Pair<UByte, R>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} // Returns a list of values built from the elements of `this` array and the [other] array with the same index * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection. 

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun <R> UShortArray.zip(other: Array<R>): List<Pair<UShort, R>> { 
    return zip(other) { t1, t2 -> t1 to t2 } 
} // Returns a list of values built from the elements of `this` array and the [other] array with the same index * using the provided [transform] function applied to each pair of elements. * The returned list has length of the shortest collection. 
```

In summary, the code defines a series of `infix` functions that zip together two arrays and return a list of pairs or values, based on the type of the arrays. The returned list has the length of the shortest collection.
built from the elements of `this` collection and [other] array with the same index.

* The returned list has length of the shortest collection.
* `@sample samples.collections.Iterables.Operations.zipIterable`

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public infix fun <R> UByteArray.zip(other:
Iterable<R>): List<Pair<UByte, R>> { `n return zip(other) { t1, t2 -> t1 to t2 }[un]**n * Returns a list of pairs built from the elements of `this` collection and [other] array with the same index.

* The returned list has length of the shortest collection.
* `@sample samples.collections.Iterables.Operations.zipIterable`

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
public infix fun <R> UShortArray.zip(other:
Iterable<R>): List<Pair<UShort, R>> { `n return zip(other) { t1, t2 -> t1 to t2 }[un]**n * Returns a list of values built from the elements of `this` array and the [other] collection with the same index
* using the provided [transform] function applied to each pair of elements.

* The returned list has length of the shortest collection.
* `@sample samples.collections.Iterables.Operations.zipIterableWithTransform`

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UIntArray.zip(other: Iterable<R>, transform: (a: UInt, b: R) -> V): List<Pair<UInt, V>> {
    val arraySize = size
    val list = ArrayList<Pair<UInt, V>>()
    var i = 0
    for (element in other) {
        list.add(transform(array[i++], element))
    }
    return list
}

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<Pair<UByte, V>> {
    val arraySize = size
    val list = ArrayList<Pair<UByte, V>>()
    var i = 0
    for (element in other) {
        list.add(transform(array[i++], element))
    }
    return list
}

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
    val arraySize = size
    val list = ArrayList<Pair<UShort, V>>()
    var i = 0
    for (element in other) {
        list.add(transform(array[i++], element))
    }
    return list
}

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
UByteArray.zip(other: Iterable<R>, transform: (a: UByte, b: R) -> V): List<Pair<UByte, V>> {
    val arraySize = size
    val list = ArrayList<Pair<UByte, V>>()
    var i = 0
    for (element in other) {
        list.add(transform(array[i++], element))
    }
    return list
}

@SinceKotlin("1.3")@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun <R, V>
ULongArray.zip(other: Iterable<R>, transform: (a: ULong, b: R) -> V): List<Pair<ULong, V>> {
    val arraySize = size
    val list = ArrayList<Pair<ULong, V>>()
    var i = 0
    for (element in other) {
        list.add(transform(array[i++], element))
    }
    return list
}
List<Pair<UByte, UByte>> { return zip(other) { t1, t2 -> t1 to t2 } }

/**
 * Returns a list of pairs built from the elements of `this` array and the [other] array with the same index.
 * The returned list has length of the shortest collection.
 * * @param samples.collections.Iterables.Operations.zipIterable
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
    return zip(other) { t1, t2 -> t1 to t2 }
}

/**
 * Returns a list of values built from the elements of `this` array and the [other] array with the same index
 * * using the provided [transform] function applied to each pair of elements.
 * * @param samples.collections.Iterables.Operations.zipIterableWithTransform
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    val size = minOf(size, other.size)
    val list = ArrayList<V>(size)
    for (i in 0 until size) {
        list.add(transform(this[i], other[i]))
    }
    return list
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UShortArray.sum(): UShort {
    var sum = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2277
fun ULongArray.sum(): ULong {
    return storage.sum().toULong()
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
constructor(inline fun UByteArray.sum(): UInt {
    return sumBy { it.toUInt() }
}

/**
 * Returns the sum of all elements in the array.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
constructor(inline fun UShortArray.sum(): UInt {
    return sumBy { it.toUInt() }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UCollectionsKt")

package kotlin.collections

/*
 * NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("UComparisonsKt")

package kotlin.comparisons

/*
 * NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
 */


/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 *
@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("URangesKt")

package kotlin.ranges

//
// NOTE: THIS FILE IS AUTO-GENERATED by the GenerateStandardLib.kt
// See: https://github.com/JetBrains/kotlin/tree/master/libraries/stdlib

import kotlin.random.*

/**
 * Returns a random element from this range.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntRange.random(): UInt {
    return random(Random)
}

/**
 * Returns a random element from this range using the specified source of randomness.
 * 
 * @throws IllegalArgumentException if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntRange.random(random: Random): UInt {
    try {
        return random.nextUInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntRange.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UInt, b: UInt, c: UInt): UInt {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: ULong, b: ULong, c: ULong): ULong {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun maxOf(a: UByte, b: UByte, c: UByte): UByte {
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the greater of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return maxOf(a, maxOf(b, c))
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UInt, b: UInt): UInt {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: ULong, b: ULong): ULong {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UByte, b: UByte): UByte {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UShort, b: UShort): UShort {
    return if (a <= b) a else b
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UInt, b: UInt, c: UInt): UInt {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: ULong, b: ULong, c: ULong): ULong {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun minOf(a: UByte, b: UByte, c: UByte): UByte {
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
    return minOf(a, minOf(b, c))
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun UIntRange.random(random: Random): UInt {
    try {
        return random.nextUInt(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns the smaller of three values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public fun ULongRange.random(random: Random): ULong {
    try {
        return random.nextULong(this)
    } catch(e: IllegalArgumentException) {
        throw NoSuchElementException(e.message)
    }
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UIntRange.randomOrNull(): UInt? {
    return randomOrNull(Random)
}

/**
 * Returns a random element from this range, or `null` if this range is empty.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun ULongRange.randomOrNull(): ULong? {
    return randomOrNull(Random)
}

/*
a random element from this range, or `null` if this range is empty.

```kotlin
public inline fun ULongRange.randomOrNull(): ULong? {
    return randomOrNull(Random())
}
```

Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.

```kotlin
public fun UIntRange.randomOrNull(random: Random): UInt? {
    if (isEmpty())
        return null
    return random.nextUInt(this)
}
```

Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.

```kotlin
public fun ULongRange.randomOrNull(random: Random): ULong? {
    if (isEmpty())
        return null
    return random.nextULong(this)
}
```

Returns a random element from this range using the specified source of randomness, or `null` if this range is empty.

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun UIntRange.contains(element: UInt?): Boolean {
    return element != null && contains(element)
}
```

Returns `true` if this range contains the specified [element].

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline operator fun ULongRange.contains(element: ULong?): Boolean {
    return element != null && contains(element)
}
```

Returns `true` if this range contains the specified [element].

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UByte.downTo(to: UByte): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

Returns a progression from this value down to the specified [to] value with the step -1.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UInt.downTo(to: UInt): UIntProgression {
    return UIntProgression.fromClosedRange(this, to, -1)
}
```

Returns a progression from this value down to the specified [to] value with the step -1.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun ULong.downTo(to: ULong): ULongProgression {
    return ULongProgression.fromClosedRange(this, to, -1L)
}
```

Returns a progression from this value down to the specified [to] value with the step -1.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public operator fun UShort.downTo(to: UShort): UIntProgression {
    return UIntProgression.fromClosedRange(this.toUInt(), to.toUInt(), -1)
}
```

Returns a progression from this value down to the specified [to] value with the step -1.
a progression that goes over the same range in the opposite direction with the same step.

```kotlin
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UIntProgression.reversed(): UIntProgression {
    return UIntProgression.fromClosedRange(last, first, -step)
}

/**
 * Returns a progression that goes over the same range in the opposite direction with the same step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULongProgression.reversed(): ULongProgression {
    return ULongProgression.fromClosedRange(last, first, -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun UIntProgression.step(step: Int): UIntProgression {
    checkStepIsPositive(step > 0, step)
    return UIntProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

/**
 * Returns a progression that goes over the same range with the given step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public infix fun ULongProgression.step(step: Long): ULongProgression {
    checkStepIsPositive(step > 0, step)
    return ULongProgression.fromClosedRange(first, last, if (this.step > 0) step else -step)
}

/**
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 * Returns a range from this value up to but excluding the specified [to] value.
 * If the [to] value is less than or equal to `this` value, then the returned range is empty.
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not greater than the specified [maximumValue].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtLeast(minimumValue: UInt): UInt {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not greater than the specified [maximumValue].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtLeast(minimumValue: ULong): ULong {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not greater than the specified [maximumValue].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtLeast(minimumValue: UByte): UByte {
    return if (this < minimumValue) minimumValue else this
}

/**
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not less than the specified [minimumValue].
 * Ensures that this value is not greater than the specified [maximumValue].
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtLeast(minimumValue: UShort): UShort {
    return if (this < minimumValue) minimumValue else this
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2281
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceAtMost(maximumValue: UInt): UInt {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceAtMost(maximumValue: ULong): ULong {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceAtMost(maximumValue: UByte): UByte {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value is not greater than the specified [maximumValue].
 * @return this value if it's less than or equal to the [maximumValue] or the [maximumValue] otherwise.
 * @sample samples.comparisons.ComparableOps.coerceAtMostUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceAtMost(maximumValue: UShort): UShort {
    return if (this > maximumValue) maximumValue else this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(minimumValue: UInt, maximumValue: UInt): UInt {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(minimumValue: ULong, maximumValue: ULong): ULong {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.coerceIn(minimumValue: UByte, maximumValue: UByte): UByte {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}

/**
 * Ensures that this value lies in the specified range [minimumValue]..[maximumValue].
 * @return this value if it's in the range, or [minimumValue] if this value is less than [minimumValue], or [maximumValue] if this value is greater than [maximumValue].
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.coerceIn(minimumValue: UShort, maximumValue: UShort): UShort {
    if (minimumValue > maximumValue) throw IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    if (this < minimumValue) return minimumValue
    if (this > maximumValue) return maximumValue
    return this
}
IllegalArgumentException("Cannot coerce value to an empty range: maximum $maximumValue is less than minimum $minimumValue.")
    } else if (this < minimumValue) return minimumValue
    else if (this > maximumValue)
        return maximumValue
    else return this
}

/**
 * Ensures that this value lies in the specified [range].
 * 
 * @return this value if it's in the [range], or 'range.start' if this value is less than 'range.start', or 'range.endInclusive' if this value is greater than 'range.endInclusive'.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.coerceIn(range: ClosedRange<UInt>): UInt {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<UInt>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Ensures that this value lies in the specified [range].
 * 
 * @return this value if it's in the [range], or 'range.start' if this value is less than 'range.start', or 'range.endInclusive' if this value is greater than 'range.endInclusive'.
 * 
 * @sample samples.comparisons.ComparableOps.coerceInUnsigned
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun ULong.coerceIn(range: ClosedRange<ULong>): ULong {
    if (range is ClosedFloatingPointRange) {
        return this.coerceIn<ULong>(range)
    }
    if (range.isEmpty()) throw IllegalArgumentException("Cannot coerce value to an empty range: $range.")
    return when {
        this < range.start -> range.start
        this > range.endInclusive -> range.endInclusive
        else -> this
    }
}

/**
 * Returns the sum of all elements in the sequence.
 * 
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUInt")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UInt>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * 
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfULong")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<ULong>.sum(): ULong {
    var sum: ULong = 0uL
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * 
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUByte")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UByte>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}

/**
 * Returns the sum of all elements in the sequence.
 * 
 * The operation is _terminal_.
 */
@kotlin.jvm.JvmName("sumOfUShort")
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Sequence<UShort>.sum(): UInt {
    var sum: UInt = 0u
    for (element in this) {
        sum += element
    }
    return sum
}
mutable iterator is invariant on its element type.

### List Interface

```java
public interface List<E> : Collection<E> {
    // Query Operations
    /**
     * Returns the index of the first occurrence of the specified
     * element in this list, or -1 if the specified
     * element is not contained in this list.
     */
    public fun indexOf(element: @UnsafeVariance E): Int

    /**
     * Returns the index of the last occurrence of the specified
     * element in this list, or -1 if the specified
     * element is not contained in this list.
     */
    public fun lastIndexOf(element: @UnsafeVariance E): Int

    /**
     * Returns the element at the specified index in the list.
     */
    public operator fun get(index: Int): E

    // Modification Operations
    /**
     * Adds all of the elements of the specified collection to
     * this collection.
     */
    public fun addAll(elements: Collection<E>): Boolean

    /**
     * Removes all of this collection's elements that are contained in the specified collection.
     * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.
     */
    public fun retainAll(elements: Collection<E>): Boolean

    /**
     * Retains only the elements in this
     * collection that are contained in the specified collection.
     * @return `true` if any of the specified elements was removed from the collection, `false` if the collection was not modified.
     */
    public fun retainAll(elements: Collection<E>): Boolean

    /**
     * Removes all elements from this collection.
     */
    public fun clear(): Unit

    // Positional Access Operations
    /**
     * Returns an iterator over the elements of this sequence that supports removing elements during iteration.
     */
    override fun iterator(): Iteratable<T>
next() = nextBoolean()eln /** Returns the next value in the sequence without boxing. */eln public abstract fun
nextBoolean(): Booleaneln

/* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file. */eln

/* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language
contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file. */

package kotlin.ranges

/**
 * A progression of values of type `Char`.
 */
public open class CharProgression

    internal constructor
    (
            start: Char,
            endInclusive: Char,
            step: Int
    }
}

/**
 * A progression of values of type `Int`.
 */
public
public fun fromClosedRange(rangeStart: Char, rangeEnd: Char, step: Int): CharProgression =

    companion object {
        /**
         * Creates CharProgression within the specified
         * bounds of a closed range.
         
         * The progression starts with the [rangeStart] value and goes toward the
         * [rangeEnd] value not excluding it, with the specified [step].
         * In order to go backwards the [step] must be negative.
         * The progression starts with the [rangeStart] value and goes toward the
         * [rangeEnd] value not excluding it, with the specified [step].
         */

    }

/**
 * A progression of values of type `Long`.
 */
public open class LongProgression

    internal constructor
    (
            start: Long,
            endInclusive: Long,
            step: Long

    }

/**
 * An iterator
 * over a progression of values of type `Long`.
 * @property step the number by which the value is incremented on each step.
 */
internal class LongProgressionIterator(first: Long, last: Long, val step: Long) : LongIterator()

    /**
     * The step of the progression.
     */
    public val step: Long = step

    override fun iterator(): LongIterator =

        Char = getProgressionLastElement(start.toInt(), endInclusive.toInt(), step).toChar()

    /**
     * The first element in the progression.
     */
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private var next = if (hasNext) first else finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextChar(): Char {
        val value = next
        if (value == finalElement)
            throw kotlin.NoSuchElementException()
        else {
            next += step

        }
        return value
    }
}

/**
 * An iterator
 * over a progression of values of type `Char`.
 * @property step the number by which the value is incremented on each step.
 */
internal class CharProgressionIterator(first: Char, last: Char, val step: Int) :

    CharIterator() {
    private val finalElement = last.toInt()
    private var hasNext: Boolean = if (step > 0) first <= finalElement

    override fun hasNext(): Boolean = hasNext

    override fun nextChar(): Char {
        val value = next
        if (value == finalElement)
            throw kotlin.NoSuchElementException()
        else {
            next += step

        }
        return value
    }
}

/**
 * An iterator
 * over a progression of values of type `Int`.
 * @property step the number by which the value is incremented on each step.
 */
internal class IntProgression

    internal constructor
    (
            start: Int,
            endInclusive: Int,
            step: Int

    }

/**
 * An iterator
 * over a progression of values of type `Int`.
 * @property step the number by which the value is incremented on each step.
 */
internal class IntProgression

    internal constructor
    (
            start: Int,
            endInclusive: Int,
            step: Int

    }

/**
 * An iterator
 * over a progression of values of type `Int`.
 * @property step the number by which the value is incremented on each step.
 */
internal class IntProgression

    internal constructor
    (
            start: Int,
            endInclusive: Int,
            step: Int

    }

/**
 * An iterator
 * over a progression of values of type `Int`.
 * @property step the number by which the value is incremented on each step.
 */
internal class IntProgression

    internal constructor
    (
            start: Int,
            endInclusive: Int,
            step: Int

    }

/**
 * An iterator
 * over a progression of values of type `Int`.
 * @property step the number by which the value is incremented on each step.
 */
internal class IntProgression

    internal constructor
    (
```kotlin
internal constructor
    (start: Long,
     endInclusive: Long,
     step: Long)
    : Iterable<Long> {
    init {
        if (step == 0L) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Long.MIN_VALUE to avoid overflow on negation.")
        *\n        public val first: Long = start
        *\n        public val last: Long = getProgressionLastElement(start.toLong(), endInclusive.toLong(), step).toLong()
        *\n        public val step: Long = step
        *\n        override fun iterator(): LongIterator = LongProgressionIterator(first, last, step)
        *\n        override fun iterator(): IntIterator = IntProgressionIterator(first, last, step)
        *\n        override fun isEmpty(): Boolean = if (step == 0) first > last else first < last
        *\n        override fun equals(other: Any?): Boolean = other is IntProgression && (isEmpty() && other.isEmpty() ||
            first == other.first && last == other.last &&
            step == other.step)
        *\n        override fun hashCode(): Int = if (isEmpty()) -1 else (31 * (31 * first + last) +
            step) *\n        companion object {
            *\n            public fun fromClosedRange(rangeStart: Int, rangeEnd: Int, step: Int): IntProgression = IntProgression(rangeStart, rangeEnd, step)
            *\n            public fun fromClosedRange(rangeStart: Long, rangeEnd: Long, step: Long): LongProgression = LongProgression(rangeStart, rangeEnd, step)
            
            *\n            public fun min(value: T, value2: T): T
                = if (value > value2) value2 else value
            
```
one value: the 'Unit' object. This type corresponds to the 'void' type in Java.

```java
public object Unit {
    override fun toString() = "kotlin.Unit"
}
```

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
 * implied. See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.annotation

/**
 * Contains the list of code elements which are the possible annotation targets
 * @property allowedTargets list of allowed annotation targets
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Target(vararg val allowedTargets: AnnotationTarget)

/**
 * Determines how an annotation is stored in binary output.
 * This meta-annotation indicates the kinds of code elements which are possible targets of an annotation.
 * If the target meta-annotation is not present on an annotation declaration, the annotation is applicable to the following elements:
 * [CLASS], [PROPERTY], [FIELD], [LOCAL_VARIABLE], [VALUE_PARAMETER], [CONSTRUCTOR], [FUNCTION], [PROPERTY_GETTER], [PROPERTY_SETTER].
 * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
public annotation class Retention(val value: AnnotationRetention = AnnotationRetention.RUNTIME)

/**
 * Determines that an annotation is a part of public API and therefore should be included in the generated documentation for the element to which the annotation is applied.
 * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class PureReifiable

/**
 * Specifies that the corresponding built-in method exists depending on platform.
 * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Repeatable

/**
 * Determines that an annotation is applicable twice or more on a single code element
 * @property value necessary annotation retention (RUNTIME, BINARY or SOURCE)
 */
@Target(AnnotationTarget.ANNOTATION_CLASS)
@MustBeDocumented
public annotation class Internal

}```
module JDK.

* For example MutableMap.remove(K, V) available only if corresponding
  'java/util/Map.remove(Ljava/lang/Object;Ljava/lang/Object;)' is defined in JDK (i.e. for major versions >= 8)

* @Target(AnnotationTarget.FUNCTION)
  @Retention(AnnotationRetention.BINARY)
  @internal annotation

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the
 * license/LICENSE.txt file. */

package kotlin.internal

// a mod b (in arithmetical sense)
private fun mod(a: Int, b: Int): Int {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

private fun mod(a: Long, b: Long): Long {
    val mod = a % b
    return if (mod >= 0) mod else mod + b
}

// (a - b) mod c
private fun differenceModulo(a: Int, b: Int, c: Int): Int {
    return mod(mod(a, c) - mod(b, c), c)
}

private fun differenceModulo(a: Long, b: Long, c: Long): Long {
    return mod(mod(a, c) - mod(b, c), c)
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in
 * the range from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 */
@PublishedApi
internal fun getProgressionLastElement(start: Int, end: Int, step: Int): Int = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the
 * progression which is in the range from [start] to [end] in case of a positive [step], or from [end] to [start] in case of a negative [step].
 */
@PublishedApi
internal fun getProgressionLastElement(start: Long, end: Long, step: Long): Long = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step)
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, -step)
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and limitations under the License.
 */

package kotlin.reflect

/**
 * Represents a parameter passed to a function or a property getter/setter, including `this` and extension receiver parameters.
 */
public interface KParameter : KAnnotatedElement {
    /**
     * 0-based index of this parameter in the parameter list of its containing callable.
     */
    public val index: Int

    /**
     * Name of this parameter as it was declared in the source code,
     * or `null` if the parameter has no name or its name is not available at runtime.
     */
    public val name: String?

    /**
     * Type of this parameter. For a `vararg` parameter, this is the type of the corresponding array,
     * not the individual element.
     */
    public val type: KType

    /**
     * Kind of this parameter.
     */
    public enum class Kind {
        INSTANCE,

        EXTENSION_RECEIVER,

        ...
value parameter. * An `VALUE` parameter is optional and can be omitted when making a call via `KCallable.callBy`, or `false` otherwise.

* A parameter is optional in any of the two cases:

  1. The default value is provided at the declaration of this parameter.
  2. The parameter is declared in a member function and one of the corresponding parameters in the super functions is optional.

```kotlin
public val isOptional: Boolean

// *true* if this parameter is `vararg`.
@SinceKotlin("1.1")
public data class KTypeProjection constructor(

    public companion object {

    /**
     * `true` if this type is not denotable in Kotlin, for example if it is an
     * intersection type.

     * @SinceKotlin("1.1")
     * public val isVararg: Boolean

    /**
     * `true` if this parameter is optional and can be omitted when
     * making a call via `KCallable.callBy`, or `false` otherwise.
     * A parameter is optional in any of the two cases:

     1. The default value is provided at the declaration of this parameter.
     2. The parameter is declared in a member function and one of the corresponding parameters in the super functions is optional.
```
KTypeProjection(KVariance.OUT, type)\n/* The affected type parameter or type is +covariant-, which means it has no variance applied to it. */
\nINVARIANT;\n/**\n * The affected type parameter or type is +contravariant-. Denoted by the `in` modifier in the source code.\n */

* @see [KTypeParameter.variance]*\n*

@SinceKotlin("1.1")\nenum class KVariance {\n    INVARIANT,\n    IN,\n    OUT,\n    /**\n     * Visibility of declarations marked with the `public` modifier, or with no modifier at all.\n     */\n    PUBLIC,\n    /**\n     * Visibility of declarations marked with the `protected` modifier.\n     */\n    PROTECTED,\n    /**\n     * Visibility of declarations marked with the `internal` modifier.\n     */\n    INTERNAL,\n    /**\n     * Visibility of declarations marked with the `private` modifier.\n     */\n    PRIVATE,\n}

arrayIterator(array: dynamic, type: String?) = when (type) {\n    null -> {\n        val arr: Array<dynamic> = array\n        object : Iterator<dynamic> {\n            var index = 0\n            override fun hasNext() = index < arr.size\n            override fun next() = if (index < arr.size) arr[index++] else throw NoSuchElementException("$index")\n        }\n    }\n    "BooleanArray" -> booleanArrayIterator(array)\n    "ByteArray" -> byteArrayIterator(array)\n    "ShortArray" -> shortArrayIterator(array)\n    "CharArray" -> charArrayIterator(array)\n    "IntArray" -> intArrayIterator(array)\n    "LongArray" -> longArrayIterator(array)\n    "FloatArray" -> floatArrayIterator(array)\n    "DoubleArray" -> doubleArrayIterator(array)\n    else -> throw IllegalStateException("Unsupported type argument for arrayIterator: $type")\n}

booleanArrayIterator(array: BooleanArray) = object : BooleanIterator {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextBoolean() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")\n}

byteArrayIterator(array: ByteArray) = object : ByteIterator {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextByte() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")\n}

shortArrayIterator(array: ShortArray) = object : ShortIterator {\n    var index = 0\n    override fun hasNext() = index < array.size\n    override fun nextShort() = if (index < array.size) array[index++] else throw NoSuchElementException("$index")\n
String =
        when (value) {
            is Failure -> value.toString() // "Failure($exception)"
            else ->
                "Success($value)"
        }

    // companion with constructors

    /**
     * Companion object for [Result] class that contains its constructor functions
     * [success] and [failure].
     */
    public companion object {

        /**
         * Returns an instance that encapsulates the given [value] as successful value.
         */
        @InlineOnly
        public inline fun <T> success(value: T): Result<T> =
            Result(value)

        /**
         * Returns an instance that encapsulates the given [Exception] as failure.
         */
        @InlineOnly
        public inline fun <T> failure(exception: Throwable): Result<T> =
            Result(createFailure(exception))

        internal class Failure(
            @JvmField
            val exception: Throwable
        ) : Serializable {
            override
            fun equals(other: Any?): Boolean = other is Failure && exception == other.exception
            override fun
            hashCode(): Int = exception.hashCode()
            override fun toString(): String = "Failure($exception)"
        }
    }

    // transformations

    /**
     * Returns the encapsulated result of the given
[transform] function applied to the encapsulated value if this instance represents [success][Result.isSuccess] or the original encapsulated [Throwable] exception if it is [failure][Result.isFailure].

* Note, that this function rethrows any [Throwable] exception thrown by [transform] function. See [mapCatching] for an alternative that encapsulates exceptions.

* *na@InlineOnly}@SinceKotlin("1.3")*/n
public inline fun <R, T> Result<T>.map(transform: (value: T) -> R): Result<R> {
    contract {
        callsInPlace(transform, InvocationKind.AT_MOST_ONCE)
    }
    return when (val isSuccess = transform(value as T)) {
        true -> Result.success(transform(value as T))
        else -> Result(value)
    }
}

/**
 * Returns the encapsulated result of the given [transform] function applied to the encapsulated value if this instance represents [success][Result.isSuccess] or the original encapsulated [Throwable] exception if it is [failure][Result.isFailure].
 * Note, that this function rethrows any [Throwable] exception thrown by [transform] function. See [mapCatching] for an alternative that encapsulates exceptions.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T> Result<T>.mapCatching(transform: (value: T) -> R): Result<R> {
    return when (val isSuccess = transform(value as T)) {
        true -> runCatching { transform(value as T) }
        else -> Result(value)
    }
}

/**
 * Returns the encapsulated result of the given [transform] function applied to the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure] or the original encapsulated value if it is [success][Result.isSuccess].
 * Note, that this function rethrows any [Throwable] exception thrown by [transform] function and encapsulates it as a failure. See [recoverCatching] for an alternative that rethrows exceptions.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.recover(transform: (exception: Throwable) -> R): Result<R> {
    contract {
        callsInPlace(transform, InvocationKind.AT_MOST_ONCE)
    }
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> Result.success(transform(exception))
    }
}

/**
 * Returns the encapsulated result of the given [transform] function applied to the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure] or the original encapsulated value if it is [success][Result.isSuccess].
 * Note, that this function rethrows any [Throwable] exception thrown by [transform] function and encapsulates it as a failure. See [recoverCatching] for an alternative that rethrows exceptions.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <R, T : R> Result<T>.recoverCatching(transform: (exception: Throwable) -> R): Result<R> {
    val value = value // workaround for inline classes BE bug
    return when (val exception = exceptionOrNull()) {
        null -> this
        else -> runCatching { transform(exception) }
    }
}

// -------------------

/**
 * Performs the given [action] on the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {
    exceptionOrNull()?.let { action(it) }
    return this
}

/**
 * Performs the given [action] on the encapsulated value if this instance represents [success][Result.isSuccess].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {
    if (isSuccess) action(value as T)
    return this
}

// -------------------

// "peek" onto value/exception and pipe

/**
 * Performs the given [action] on the encapsulated [Throwable] exception if this instance represents [failure][Result.isFailure].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onFailure(action: (exception: Throwable) -> Unit): Result<T> {
    exceptionOrNull()?.let { action(it) }
    return this
}

/**
 * Performs the given [action] on the encapsulated value if this instance represents [success][Result.isSuccess].
 * Returns the original `Result` unchanged.
 */
@InlineOnly
@SinceKotlin("1.3")
public inline fun <T> Result<T>.onSuccess(action: (value: T) -> Unit): Result<T> {
    if (isSuccess) action(value as T)
    return this
}

// -------------------

// -------------------

// -------------------

// -------------------

// -------------------

// -------------------


blic annotation class RestrictsSuspension

/**
 * Resumes the execution of the corresponding coroutine passing [value] as the return value of the last suspension point.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation<T>.resume(value: T): Unit =
    resumeWith(Result.success(value))

/**
 * Resumes the execution of the corresponding coroutine so that the [exception] is re-thrown right after the
 * last suspension point.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T> Continuation<T>.resumeWithException(exception: Throwable): Unit =
    resumeWith(Result.failure(exception))

/**
 * Creates a [Continuation] instance with the given [context] and implementation of [resumeWith] method.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun <T, C : CoroutineContext> Continuation(
    context: C,
    crossinline resumeWith: (Result<T>) -> Unit
): Continuation<T> =
    object : Continuation<T> {
        override val context: CoroutineContext
            get() = context

        override fun resumeWith(result: Result<T>) =
            resumeWith(result)
    }

/**
 * Creates a coroutine without a receiver and with result type [T].
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <T> (suspend () -> T).createCoroutine(
    completion: Continuation<T>
): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(completion).intercepted(), COROUTINE_SUSPENDED)

/**
 * Creates a coroutine with receiver type [R] and result type [T].
 */
@SinceKotlin("1.3")
@Suppress("UNCHECKED_CAST")
public fun <R, T> (suspend R.() -> T).createCoroutine(
    receiver: R,
    completion: Continuation<T>
): Continuation<Unit> =
    SafeContinuation(createCoroutineUnintercepted(receiver, completion).intercepted(), COROUTINE_SUSPENDED)

/**
 * Obtains the current
 */
@SinceKotlin("1.3")
@Suppress("WRONG_MODIFIER_TARGET")
@InlineOnly
public suspend inline val coroutineContext: CoroutineContext
    get() {
        throw NotImplementedError("Implemented as intrinsic")
    }

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

Open Source Used In IOx-FogDirector-AC18 1.17.0 2298
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.coroutines.intrinsics

import kotlin.coroutines.*
import kotlin.internal.InlineOnly

/**
 * Starts an unintercepted coroutine without a receiver and with result type [T] and executes it until its first suspension.
 * Returns the result of the coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.
 * In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.
 * The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might
 * be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation
 * context is established.
 * This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of
 * the suspended coroutine using a reference to the suspending function.

@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <T> (suspend () -> T).startCoroutineUninterceptedOrReturn(
    completion: Continuation<T>.
): Any? =
    this.asDynamic()(completion, false)

/**
 * Starts an unintercepted coroutine with receiver type [R] and result type [T] and executes it until its first suspension.
 * Returns the result of the coroutine or throws its exception if it does not suspend or [COROUTINE_SUSPENDED] if it suspends.
 * In the latter case, the [completion] continuation is invoked when the coroutine completes with a result or an exception.
 * The coroutine is started directly in the invoker's thread without going through the [ContinuationInterceptor] that might
 * be present in the completion's [CoroutineContext]. It is the invoker's responsibility to ensure that a proper invocation
 * context is established.
 * This function is designed to be used from inside of [suspendCoroutineUninterceptedOrReturn] to resume the execution of
 * the suspended coroutine using a reference to the suspending function.

@SinceKotlin("1.3")
@InlineOnly
public actual inline fun <R, T> (suspend R.() -> T).startCoroutineUninterceptedOrReturn(
    receiver: R,
    completion: Continuation<T>.
): Any? =
    this.asDynamic()(receiver, completion, false)

/**
 * Creates unintercepted coroutine without receiver and with result type [T].
 * This function creates a new, fresh instance of suspensible computation every time it is invoked.
 * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.
 * Note that [completion] of this function may get invoked in an arbitrary context.
 * [Continuation.intercepted] can be used to acquire the intercepted continuation.
 * Invocation of `resume(Unit)` on intercepted continuation guarantees that execution of both the coroutine and [completion] happens in the invocation context established by
 * [ContinuationInterceptor].

@SinceKotlin("1.3")
public actual fun <T> (suspend () -> T).createCoroutineUnintercepted(
    completion: Continuation<T>.
): Continuation<Unit> =
    this.asDynamic().length == 2 {
        this.asDynamic()(completion, true)
    } else {
        createCoroutineFromSuspendFunction(completion) {
            this.asDynamic().completion
        }
    }

/**
 * Creates unintercepted coroutine with receiver type [R] and result type [T].
 * This function creates a new, fresh instance of suspensible computation every time it is invoked.
 * To start executing the created coroutine, invoke `resume(Unit)` on the returned [Continuation] instance.
 * The [completion] continuation is invoked when the coroutine completes with result or exception.
 * This function returns unintercepted continuation.

@SinceKotlin("1.3")
public actual fun <R, T> (suspend R.() -> T).createCoroutineUnintercepted(
    receiver: R,
): Continuation<T>.
 =
{ // When `suspended` is true the continuation is created, but not executed
    this.asDynamic().length == 2 {
        // This function creates a new, fresh instance of suspensible computation every time it is invoked.
        this.asDynamic().completion
    }
}
coroutine and [completion] happens in the invocation context established by\n * *\n * [ContinuationInterceptor].\n *\n * Repeated invocation of any resume function on the resulting continuation corrupts the\n * state machine of the coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n *\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n * */\n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if\n * (this.asDynamic().length == 3) \{ \n * // When \`suspended\` is true the continuation is created, but not executed\n * this.asDynamic()(receiver, completion, true)\n * } else \{ \n * createCoroutineFromSuspendFunction(completion)\n * \} \n *\n * Repeated invocation of any resume function on the resulting continuation corrupts\n * the state machine of the\n * coroutine and may result in arbitrary behaviour or exception.\n * *\n * @SinceKotlin("1.3")\n * public actual fun <R, T>\n * (suspend R.() -> T).createCoroutineUnintercepted(\n * receiver: R;\n * completion: Continuation<T>\n * )\n * Continuation<Unit> =\n * // Kotlin/JS suspend lambdas have an extra parameter \`suspended\`\n * if

---

Open Source Used In IOX-FogDirector-AC18 1.17.0 2300

---
public actual fun Float.toBits(): Int =
definedExternally

/**
 * Returns a bit representation of the specified floating-point value as [Int]
 * according to the IEEE 754 floating-point "single format" bit layout, preserving 'NaN' values exact layout.
 * @since Kotlin("1.2")
 * @library("floatToBits")
 * public actual fun Float.toBits(): Int =
definedExternally

/**
 * Returns the [Float] value corresponding to a given bit representation.
 * @sinceKotlin("1.2")
 * @library("floatToRawBits")
 * public actual fun Float.toRawBits(): Int =
definedExternally

/**
 * Returns a string representation of this [Long] value in the specified [radix].
 * @sinceKotlin("1.2")
 * @library("text")
 * @sample samples.collections.Collections.Elements.toString
 * public actual fun Long.toString(radix: Int): String =

package kotlin.collections

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Reverses elements in the list in-place.
 */
public actual fun <T> MutableList<T>.reverse(): Unit {
    val midPoint = (size / 2) - 1
    if (midPoint < 0) return
    var reverseIndex = lastIndex
    for (index in 0..midPoint) {
        val tmp = this[index]
        this[index] = this[reverseIndex]
        this[reverseIndex] = tmp
        reverseIndex--
    }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

import kotlin.js.*

/**
 * Returns a character at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this char sequence.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun CharSequence.elementAt(index: Int): Char {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, length: $length") }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun UIntArray.elementAt(index: Int): UInt {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun ULongArray.elementAt(index: Int): ULong {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun UByteArray.elementAt(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun UShortArray.elementAt(index: Int): UShort {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.collections

import kotlin.js.*
import kotlin.ranges.contains
import kotlin.ranges.reversed

/**
 * Returns an element at the given [index] or throws an [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 * @sample samples.collections.Collections.Elements.elementAt
 */
public actual fun IntArray.elementAt(index: Int): Int {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}
/**
 * Returns an element at the given [index] or throws a [IndexOutOfBoundsException] if the [index] is out of bounds of this array.
 */

public actual fun UByteArray.elementAtOrElse(index: Int): UByte {
    return elementAtOrElse(index) { throw IndexOutOfBoundsException("index: $index, size: $size") }
}

/**
 * Returns a [List] that wraps the original array.
 */

public actual fun UByteArray.asList(): List<UByte> {
    return object : AbstractList<UByte>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UByte): Boolean = this@asList.contains(element)
        override fun get(index: Int): UByte {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UByte): Int {
            if ((element as Any?) !is UByte) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}

/**
 * Returns a [List] that wraps the original array.
 */

public actual fun UShortArray.asList(): List<UShort> {
    return object : AbstractList<UShort>(), RandomAccess {
        override val size: Int get() = this@asList.size
        override fun isEmpty(): Boolean = this@asList.isEmpty()
        override fun contains(element: UShort): Boolean = this@asList.contains(element)
        override fun get(index: Int): UShort {
            AbstractList.checkElementIndex(index, size)
            return this@asList[index]
        }
        override fun indexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.indexOf(element)
        }
        override fun lastIndexOf(element: UShort): Int {
            if ((element as Any?) !is UShort) return -1
            return this@asList.lastIndexOf(element)
        }
    }
}
String): JQuery = asDynamic().dialog(mode)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.dialog(mode: String, param: String, value: Any?): JQuery = asDynamic().dialog(mode, param, value)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.button(): JQuery = asDynamic().button()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.accordion(): JQuery = asDynamic().accordion()

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.draggable(params: Json): JQuery = asDynamic().draggable(params)

@Deprecated("Use declarations from 'https://bintray.com/kotlin/js-externals/kotlin-js-jquery' package instead.")
public inline fun JQuery.selectable(): JQuery = asDynamic().selectable()

@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, VALUE_PARAMETER, PROPERTY_GETTER, PROPERTY_SETTER)
@Deprecated("Use `external` modifier instead", level = DeprecationLevel.ERROR)
public annotation class native(public val name: String = "")

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeGetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeSetter

@Target(FUNCTION)
@Deprecated("Use inline extension function with body using dynamic")
public annotation class nativeInvoke

@Target(CLASS, FUNCTION, PROPERTY)
internal annotation class library(public val name: String = "")

@Target(CLASS)
internal annotation class marker

/**
 * Gives a declaration (a function, a property or a class) specific name in JavaScript.
 *
 * This may be useful in the following cases:
 *
 *   * There are two functions for which the compiler gives same name in JavaScript, you can mark one with `@JsName(...)` to prevent the compiler from reporting error.
 *   * You are writing a JavaScript library in Kotlin. The compiler produces mangled names for functions with parameters, which is unnatural for usual JavaScript developer.
 *   * For some reason you want to rename declaration, e.g. there's common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 *
 * You can put `@JsName(...)` on functions you want to be available from JavaScript.
 * For some reason you want to rename declaration, e.g. there's common term in JavaScript for a concept provided by the declaration, which in uncommon in Kotlin.
 *
 * @property name the name which compiler uses both for declaration itself and for all references to the declaration.
 * It's required to denote a valid JavaScript identifier.
 * @see JsNonModule
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, FUNCTION, PROPERTY, CONSTRUCTOR, PROPERTY_GETTER, PROPERTY_SETTER)
public annotation class JsName(actual val name: String)

/**
 * Denotes an `external` declaration that must be imported from native JavaScript library.
 * It's required to denote a valid JavaScript identifier.
 * @see JsNonModule
 */
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsModule(val import: String)

/**
 * Denotes an `external` declaration that can be used without module system.
 * It is not interpreted by the Kotlin compiler, it's passed as is directly to the target module system.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * It is not interpreted by the Kotlin compiler, it's passed as is directly to the target module system.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 * By applying `@JsModule(...)` on a file you tell the compiler to import a JavaScript object that contain all the declarations from the file.
 */

open class Person(val name: String) {
    fun hello() {
        println("Hello $name!")
    }

    fun helloWithGreeting(greeting: String) {
        println("$greeting $name!")
    }
}

println("Hello ${Person("John").name}!")
println("Hello ${Person("Jane").name}!")

Some JavaScript libraries are distributed both as a standalone downloadable piece of JavaScript and as a module available as an npm package. To tell the Kotlin compiler to accept both cases, you can augment [JsModule] with the `@JsNonModule` annotation. For example:

```kotlin
@JsModule("jquery")
@JsNonModule
@JsName("$")
external abstract class JQuery() {
    // some declarations here
    fun JQuery(element: Element): JQuery
}
```

To tell the Kotlin compiler to accept both cases, you can augment `[JsModule]` with the `@JsNonModule` annotation.

```kotlin
val @Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
val public annotation class JsNonModule

/**
 * Adds prefix to `external` declarations in a source file.
 * JavaScript does not have concept of packages (namespaces). They are usually emulated by nested objects. The compiler turns references to `external` declarations either to plain unprefixed names (in case of `plain` modules) or to plain imports. However, if a JavaScript library provides its declarations in packages, you won’t be satisfied with this.
 * You can tell the compiler to generate additional prefix before references to `external` declarations using the `@JsQualifier(...)` annotation.
 * @file:JsQualifier("my.jsPackageName")
 *
 * ```
 * package some.kotlinPackage
 * external fun foo(x: Int)
 * ```
 * @property value the qualifier to add to the declarations in the generated code. It must be a sequence of valid JavaScript identifiers separated by the `.` character. Examples of valid qualifiers are: `foo`, `bar.Baz`, `_.`. The `@see JsModule` annotation.

```kotlin
@Target(AnnotationTarget.FILE)
public annotation class JsQualifier
```

```kotlin
@ExperimentalJsExport
```

```kotlin
@Retention(AnnotationRetention.BINARY)
@Target(CLASS, PROPERTY, FUNCTION, FILE)
public annotation class JsExport
```

```
package kotlin.jvm

// TODO: find how to deprecate these

@Target(AnnotationTarget.FIELD)
public actual annotation class Volatile

@Target(AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY_GETTER, AnnotationTarget.PROPERTY_SETTER)
public actual annotation class Synchronized
```

```
package kotlin.collections

/**
 * Provides a skeletal implementation of the `[MutableCollection]` interface.
 * The collection is invariant on its element type. Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. These are used in common generated code in stdlib.
 * TODO: find how to deprecate these ones
 *
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. These are used in common generated code in stdlib.
 */

```
fromIndex: Int, toIndex: Int) : AbstractMutableList<E>(), RandomAccess {
        private var _size: Int = 0

    init {
        AbstractList.checkRangeIndexes(fromIndex, toIndex, list.size)
        this._size = toIndex - fromIndex
    }

    override fun add(index: Int, element: E) {
        AbstractList.checkPositionIndex(index, _size)
        list.add(fromIndex + index, element)
        _size++
    }

    override fun get(index: Int): E {
        AbstractList.checkElementIndex(index, _size)
        return list[fromIndex + index]
    }

    override fun removeAt(index: Int): E {
        AbstractList.checkElementIndex(index, _size)
        return list.removeAt(fromIndex + index)
    }

    override val size: Int get() = _size
}

package kotlin.collections

/**
 * Provides a skeletal implementation of the [MutableMap] interface.
 *
 * The implementor is required to implement [entries] property, which should return mutable set of map entries,
 * and [put] function.
 */
public actual abstract class AbstractMutableMap<K, V>
    protected actual constructor() : AbstractMap<K, V>(), MutableMap<K, V> {

    /**
     */
    internal open class SimpleEntry<K, V>(override
        val key: K, value: V) : MutableMap.MutableEntry<K, V> {
        constructor(entry: Map.Entry<K, V>) :
            this(entry.key, entry.value)

        private var _value = value

        override val value: V get() = _value

        override fun setValue(newValue: V): V {
            val oldValue = this._value
            this._value = newValue
            return oldValue
        }

        override fun hashCode(): Int = entryHashCode(this)
        override fun toString(): String = entryToString(this)
        override fun equals(other: Any?): Boolean = entryEquals(this, other)

    }

    actual override fun clear() {
        entries.clear()
    }

    private var _keys: MutableSet<K>? = null
    actual override val keys: MutableSet<K>
        get() {
            if (_keys == null) {
                _keys = object :
                    AbstractMutableSet<K>() {
                        override fun add(element: K): Boolean = throw
                            UnsupportedOperationException("Add is not supported on keys")
                        override fun clear() =
                            this@AbstractMutableMap.clear()
                        override operator fun contains(element: K): Boolean = containsKey(element)
                        override fun remove(element: K): Boolean {
                            if (_keys == null) {
                                _keys = object :
                                    AbstractMutableSet<K>() {
                                        override fun add(element: K): Boolean = throw
                                            UnsupportedOperationException("Add is not supported on keys")
                                        override fun clear() =
                                            this@AbstractMutableMap.clear()
                                        override operator fun contains(element: K): Boolean = containsKey(element)
                                        override fun remove(element: K): Boolean = throw
                                            UnsupportedOperationException("Add is not supported on values")
                                        override fun get(key: K, value: V): V? = null
                                        override fun putAll(from: Map<out K, V>) {
                                            for ((key, value) in from) {
                                                put(key, value)
                                            }
                                        }
                                        override fun remove(element: K) = throw
                                            UnsupportedOperationException("Add is not supported on values")
                                        override fun contains(element: K): Boolean = containsKey(element)
                                        override val entryIterator = entries.iterator()
                                        override fun remove(element: K): Boolean {
                                            if (containsKey(element)) {
                                                return true
                                            }
                                            val key = entryIterator.next().key
                                            override fun remove() = entryIterator.remove()
                                        }
                                        override fun get(key: K, value: V): V? = null
                                        override fun put(key: K, value: V): V? = null
                                        override fun contains(key: K): Boolean = containsKey(key)
                                        override fun containsValue(value: V): Boolean = containsValue(value)
                                        override fun remove(key: K): Boolean = throw
                                        override fun putIfAbsent(key: K, value: V): Boolean =
                                            put(key, value)
                                        override fun removeIf(key: K, value: V): Boolean = throw
                                            UnsupportedOperationException("Add is not supported on values")
                                            UnsupportedOperationException("Add is not supported on values")
                                    }
                                }
                                _keys = _keys!!
                            }
                            return _keys!!
                        }
                        override fun iterator(): Iterator<K> {
                            val entryIterator = entries.iterator()
                            return object : Iterator<K> {
                                override fun hasNext(): Boolean
                                    return entryIterator.hasNext()
                                override fun next(): K = entryIterator.next().key
                                override fun remove() = entryIterator.remove()
                            }
                        }
                    }
                }
            }
            return _keys!!
        }

    actual abstract override fun put(key: K, value: V): V?

    actual override fun putAll(from: Map<out K, V>) {
        for ((key, value) in from) {
            put(key, value)
        }
    }

    private var _values: MutableCollection<V>? = null
    actual override val values: MutableCollection<V>
        get() {
            if (_values == null) {
                _values = object :
                    AbstractMutableCollection<V>() {
                        override fun add(element: V): Boolean = throw
                            UnsupportedOperationException("Add is not supported on values")
                        override fun clear() =
                            this@AbstractMutableMap.clear()
                        override operator fun contains(element: V): Boolean = containsValue(element)
                        override fun remove(element: V): Boolean = throw
                            UnsupportedOperationException("Add is not supported on values")
                        override fun get(key: K, value: V): V? = null
                        override fun putAll(from: Map<out V, K>) {
                            for ((value, key) in from) {
                                put(value, key)
                            }
                        }
                        override fun remove(element: V) = throw
                            UnsupportedOperationException("Add is not supported on values")
                        override fun contains(element: V): Boolean = containsValue(element)
                        override val entryIterator = entries.iterator()
                        override fun remove(element: V): Boolean {
                            if (containsValue(element)) {
                                return true
                            }
                            val value = entryIterator.next().value
                            override fun remove() = entryIterator.remove()
                        }
                        override fun get(key: K, value: V): V? = null
                        override fun put(key: K, value: V): V? = null
                        override fun contains(key: V): Boolean = containsValue(key)
                        override fun containsValue(value: V): Boolean = containsValue(value)
                        override val entryIterator = entries.iterator()
                        override fun remove(key: V): Boolean = throw
                            UnsupportedOperationException("Add is not supported on values")
                        override fun putIfAbsent(key: V, value: K): Boolean =
                            put(value, key)
                        override fun removeIf(key: V, value: K): Boolean = throw
                            UnsupportedOperationException("Add is not supported on values")
                        override fun replace(key: V, old: K, new: K): Pair<V, K> = throw
                            UnsupportedOperationException("Add is not supported on values")
                    }
                }
            }
            return _values!!
        }

    actual override fun remove(key: K): V? {
        AbstractMutableMap<K, V>.override fun remove(key: K): V? = null
        actual override fun putAll(from: Map<out K, V>) {
            for ((key, value) in from) {
                put(key, value)
            }
        }
        actual override fun remove(element: K) = throw
            UnsupportedOperationException("Add is not supported on values")
        actual override fun putIfAbsent(key: K, value: V): Boolean =
            put(key, value)
        actual override fun removeIf(key: K, value: V): Boolean = throw
            UnsupportedOperationException("Add is not supported on values")
            UnsupportedOperationException("Add is not supported on values")
    }

    override fun contains(element: K): Boolean = containsKey(element)
    override fun containsValue(element: V): Boolean = containsValue(element)
    override val entryIterator = entries.iterator()
    override fun remove(element: K): Boolean = throw
        UnsupportedOperationException("Add is not supported on values")
    override fun putIfAbsent(key: K, value: V): Boolean =
        put(value, key)
    override fun removeIf(key: K, value: V): Boolean = throw
        UnsupportedOperationException("Add is not supported on values")
        UnsupportedOperationException("Add is not supported on values")
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2306
if (this === other) return true
else if (other !is Collection<*>) return false

return AbstractList.orderedEquals(this, other)

override fun hashCode(): Int =
AbstractList.orderedHashCode(this)

override fun remove(key: K): V? {
    val iter = entries.iterator()
    while (iter.hasNext()) {
        val entry = iter.next()
        val k = entry.key
        if (key == k) {
            val value = entry.value
            iter.remove()
            return value
        }
    }
    return null
}


removeRange(fromIndex: Int, toIndex: Int) {  
    // ...  
    implicit override fun clear() {  
        array = emptyArray()  
        modCount++  
    }  

    actual override fun indexOf(element: E): Int = array.indexOf(element)  

    actual override fun lastIndexOf(element: E): Int = array.lastIndexOf(element)  

    override fun toString() = arrayToString(array)  

    private fun rangeCheck(index: Int) = index.apply {  
        AbstractList.checkElementIndex(index, size)  
    }  

    private fun insertionRangeCheck(index: Int) = index.apply {  
        AbstractList.checkPositionIndex(index, size)  
    }  

    sortArrayWith(array: Array<out T>, comparator: (T, T) -> Int) {  
        // ...  
    }  

    sortArrayWith(array: Array<out T>, comparator: Comparator<in T>) {  
        // ...  
    }  

    sortArray(array: Array<out T>) {  
        // ...  
    }  

    private var _stableSortingIsSupported: Boolean? = null  
    private fun getStableSortingIsSupported(): Boolean {  
        // ...  
    }  

    private fun <T> mergeSort(array: Array<T>, start: Int, endInclusive: Int, comparator: Comparator<in T>): Array<T> {  
        // ...  
    }  

    @UseExperimental(ExperimentalUnsignedTypes::class)  
    @SinceKotlin("1.3")  
    @kotlin.js.JsName("contentDeepHashCodeImpl")  
    internal fun <T> Array<out T>.contentDeepHashCodeImpl(): Int {  
        var result = 1  
        for (element in this) {  
            val elementHash = when {  
                element == null -> 0  
                isArrayish(element) -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()  
                element is UByteArray -> element.contentHashCode()  
                element is UShortArray -> element.contentHashCode()  
                element is UIntArray -> element.contentHashCode()  
                element is ULongArray -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()  
                else -> element.contentHashCode()  
            }  
            result = hashCombine(result, elementHash)  
        }  
        return result  
    }  

    @UseExperimental(ExperimentalUnsignedTypes::class)  
    @SinceKotlin("1.3")  
    @kotlin.js.JsName("hashImpl")  
    internal fun <T> Array<out T>.hashImpl(): Int {  
        var result = 0  
        for (element in this) {  
            val elementHash = when {  
                element == null -> 0  
                isArrayish(element) -> (element.unsafeCast<Array<*>>()).hashImpl()  
                element is UByteArray -> element.contentHashCode()  
                element is UShortArray -> element.contentHashCode()  
                element is UIntArray -> element.contentHashCode()  
                element is ULongArray -> (element.unsafeCast<Array<*>>()).contentDeepHashCodeImpl()  
                else -> element.contentHashCode()  
            }  
            result = hashCombine(result, elementHash)  
        }  
        return result  
    }
```kotlin
public fun <V> stringMapOf(vararg pairs: Pair<String, V>): HashMap<String, V> {
    return MutableMap<K, V>(),
    pairs.map { it.first to it.second }
}
```
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* Based on GWT HashSet

* *

/**
 * The implementation of the [MutableSet] interface, backed by a [HashMap] instance.
 */

package kotlin.collections

/**
 * The implementation of the [MutableSet] interface, backed by a [HashMap] instance.
 */
public actual open class HashSet<E> : AbstractMutableSet<E>, MutableSet<E> {

    private val map: HashMap<E, Any>

    /**
     * Constructs a new empty [HashSet].
     */
    actual constructor() {
        map = HashMap<E, Any>()
    }

    /**
     * Constructs a new [HashSet] filled with the elements of the specified collection.
     */
    actual constructor(elements: Collection<E>) {
        map = HashMap<E, Any>(elements.size)
        addAll(elements)
    }

    /**
     * Constructs a new empty [HashSet].
     *
     * @param initialCapacity the initial capacity (ignored)
     * @param loadFactor the load factor (ignored)
     *
     * @throws IllegalArgumentException if the initial capacity or load factor are negative
     */
    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    actual constructor(initialCapacity: Int, loadFactor: Float = 0.0f) {
        map = HashMap<E, Any>(initialCapacity, loadFactor)
    }

    actual constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

    /**
     * Protected constructor to specify the underlying map. This is used by
     * LinkedHashSet.
     *
     * @param map underlying map to use.
     */
    internal constructor(map: HashMap<E, Any>) {
        this.map = map
    }

    actual override fun add(element: E): Boolean {
        val old = map.put(element, this)
        return old == null
    }

    actual override fun clear() {
        map.clear()
    }

    actual override operator fun contains(element: E): Boolean = map.containsKey(element)

    actual override fun isEmpty(): Boolean = map.isEmpty()

    actual override fun iterator(): MutableIterator<E> =
        map.keys.iterator()

    actual override fun remove(element: E): Boolean = map.remove(element) != null

    actual override val size: Int get() = map.size

}

/**
 * Creates a new instance of the specialized implementation of [HashSet] with the specified [String] elements,
 * which elements the keys as properties of JS object without hashing them.
 */
public fun stringSetOf(vararg elements: String): HashSet<String> {
    return HashSet(stringMapOf<Any>()).apply {
        addAll(elements)
    }
}
(equality.equals(entry.key, key)) {
  jsDeleteProperty(backingMap, hashCode)
  size--
}
else {
  return null
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}
}.
K=String, the K type is not fixed to String statically. # because we want to have it erased to Any? in order not to generate type-safe override bridges for
* [get], [contains], [remove] etc, if they ever are generated. # *uninternal class InternalStringMap<K, V>(override val equality: EqualityComparator) : InternalMap<K, V> {

private var backingMap: dynamic = createJsMap() override var backin

private override var value: V? {

backin

private val size: Int = 0

private val size: Int = 0

override operator fun contains(key: K): Boolean {

if (key !is String) return false

val value = backingMap[key]

if (value !== undefined) {
    return true
}

val value = backingMap[key]

if (value !== undefined) {
    return true
}

override operator fun get(key: K): V? {

if (key !is String) return null

val value = backingMap[key]

if (value !== undefined) {
    return value.unsafeCast<V>()
}

override operator fun put(key: K, value: V): V? {

require(key is String)

val oldValue = backingMap[key]

backingMap[key] = value

if (oldValue === undefined) {
    size++
    structureChanged(host)
    return null
}

else {
    valueMod++
    return oldValue.unsafeCast<V>()
            setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()

override fun clear() {
    backingMap = createJsMap()
    size = 0
}

override fun iterator(): MutableIterator<MutableEntry<K, V>> {
    return object :
    } private fun newMapEntry(key: K): MutableEntry<K, V> = object : MutableEntry<K, V> {
        override val key: K get() = key
        override val value: V get() = this@InternalStringMap[key].unsafeCast<V>()

        override fun setValue(newValue: V): V = this@InternalStringMap.put(key, newValue).unsafeCast<V>()

        override fun hashCode(): Int = AbstractMap.entryHashCode(this)
        override fun equals(other: Any?): Boolean = AbstractMap.entryEquals(this, other)

        override fun toString(): String = AbstractMap.entryToString(this)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 *
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
/*
 * Based on GWT LinkedHashMap
 * Copyright 2008 Google Inc.
 */
	package kotlin.collections

import kotlin.collections.MutableMap.MutableEntry

/**
 * Hash table based implementation of the [MutableMap] interface, which additionally preserves the insertion order of entries during the iteration. The insertion order is preserved by maintaining a doubly-linked list of all of its entries. 
 * niece
 */
	public actual open class LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {

    private class ChainEntry<K, V>(key: K, value: V) : AbstractMutableMap.SimpleEntry<K, V>(key, value) {

        internal var prev: ChainEntry<K, V>? = null
        internal var next: ChainEntry<K, V>? = null
        internal var prev: ChainEntry<K, V>? = null
        internal var next: ChainEntry<K, V>? = null

        internal override var value: V? = null

        internal override var prev: ChainEntry<K, V>? = null

        internal override var next: ChainEntry<K, V>? = null

    }

    private inner class EntrySet : AbstractMutableSet<MutableEntry<K, V>>() {

        private inner class EntryIterator :

            // The last entry that was returned from this iterator.

            // The next entry to return from this iterator.

            // The previous entry that we just returned.

            // The next entry to return from this iterator.

            // The previous entry that we just returned.

        }

        /*
        * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
        *
        * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
        */
        /*
        * Based on GWT LinkedHashMap
        * Copyright 2008 Google Inc.
        */
        
        kotlin.collections

        import kotlin.collections.MutableMap.MutableEntry

        * Hash table based implementation of the [MutableMap] interface, which additionally preserves the insertion order of entries during the iteration. The insertion order is preserved by maintaining a doubly-linked list of all of its entries.

        /**
         * niece
         */

        public actual open class LinkedHashMap<K, V> : HashMap<K, V>, MutableMap<K, V> {

            override fun contains(key: K): Boolean {

                if (key !is String) return false

                val value = backingMap[key]

                if (value !== undefined) {
                    return true
                }

                val value = backingMap[key]

                if (value !== undefined) {
                    return true
                }

            }

            override fun get(key: K): V? {

                if (key !is String) return null

                val value = backingMap[key]

                if (value !== undefined) {
                    return value.unsafeCast<V>()
                }

            }

            override fun put(key: K, value: V): V? {

                require(key is String)

                val oldValue = backingMap[key]

                backingMap[key] = value

                if (oldValue === undefined) {
                    size++
                    structureChanged(host)
                    return null
                }

                else {
                    valueMod++
                    return oldValue.unsafeCast<V>()
                }

            }

            override fun clear() {
                backingMap = createJsMap()
                size = 0
            }

            override fun iterator(): MutableIterator<MutableEntry<K, V>> {
                return object :
            }
var next: ChainEntry<K, V>? = null
init {
    next = head
}

recordLastKnownStructure(map, this)
fun override fun hasNext(): Boolean {
    return next !== null
}

checkStructuralChange(map, this)
val current = next!!
val last = current
next = current.next.takeIf { it !== head }
return current

override fun remove() {
    check(last !== null)

    checkStructuralChange(map, this)

    if (!hasNext()) throw NoSuchElementException()

    val current = next!!
    last = current
    next = current.next.takeIf { it !== head }
    return current
}

override fun add(element: MutableEntry<K, V>): Boolean = throw UnsupportedOperationException("Add is not supported on entries")

override fun clear() {
    this@LinkedHashMap.clear()
}

operator fun contains(element: MutableEntry<K, V>): Boolean = containsEntry(element)

operator fun iterator(): MutableIterator<MutableEntry<K, V>> = EntryIterator()

override fun remove(element: MutableEntry<K, V>): Boolean {
    if (contains(element)) {
        this@LinkedHashMap.remove(element.key)
        return true
    } else {
        return false
    }
}

internal constructor(backingMap: HashMap<K, Any>) : super() {
    @Suppress("UNCHECKED_CAST") // expected to work due to erasure
    map = backingMap as HashMap<K, ChainEntry<K, V>>
}

constructor(initialCapacity: Int, loadFactor: Float = 0.0f) : super(initialCapacity, loadFactor) {
    map = HashMap<K, ChainEntry<K, V>>()
}

constructor(initialCapacity: Int) : this(initialCapacity, 0.0f)

constructor(original: Map<out K, V>) {
    this.putAll(original)
    map = HashMap<K, ChainEntry<K, V>>()
}

internal constructor(backingMap: HashMap<K, Any>) : super() {
    @Suppress("UNCHECKED_CAST") // expected to work due to erasure
    map = backingMap as HashMap<K, ChainEntry<K, V>>
}

constructor(original: Map<out K, V>, backingMap: HashMap<K, Any>) : super() {
    this.putAll(original)
    map = HashMap<K, ChainEntry<K, V>>()
}

internal override fun createEntrySet(): MutableList<MutableEntry<K, V>> = EntrySet()
actual override fun remove(key: K): V? {
    val entry = map.remove(key)
    if (entry != null)
        entry.remove()
    return entry.value
}

actual override fun remove(key: K): V? {
        val entry = map.remove(key)
        if (entry != null)
            entry.remove()
            return entry.value
        return null
    }

actual override val size: Int get() = map.size

/**
 * Constructs the specialized implementation of [LinkedHashMap] with [String] keys, which stores the keys as properties of JS object without hashing them.
 * */
linkedStringMapOf(vararg pairs: Pair<String, V>): LinkedHashMap<String, V> {
    //...
override fun print(message: Any?) {
    var s = String(message)
    val i = s.nativeLastIndexOf("\n", 0)
    if (i != 0) {
        buffer += s.substring(0, i)
        flush()
        s = s.substring(i + 1)
    }
    buffer += s
}

override fun flush() {
    console.log(buffer)
    buffer = ""
}

/** JsName
used to make the declaration available outside of module to test it and use at try.kotl.in
*/
@JsName("output")
internal var output = run {
    val isNode: Boolean = js("typeof process !== 'undefined'
        && process.versions
        && !!process.versions.node")
    if (isNode) NodeJsOutput(js("process.stdout")) else
        BUFFEREDOUTPUTTOCONSOLELOG().internal.InlineOnly
            .unprivece infine fun String(value: Any?): String =
                js("String!(value)\n")
            .unprivece infine fun println(message: Any?):
                String =
                    js("String!(value)\n")
            .unprivece infine fun println(message: Any?):
                String =
                    js("String!(value)\n")
fun getUTCSeconds(): Int

    public fun toDateString(): String

    public fun toISOString(): String

    public fun toJSON(): Json

    public fun toLocaleDateString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String

    public fun toLocaleDateString(locales: String, options: LocaleOptions = definedExternally): String

    public fun toLocaleString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String

    public fun toLocaleString(locales: String, options: LocaleOptions = definedExternally): String

    public fun toLocaleTimeString(locales: Array<String> = definedExternally, options: LocaleOptions = definedExternally): String

    public fun toLocaleTimeString(locales: String, options: LocaleOptions = definedExternally): String

    public fun toTimeString(): String

    public fun toUTCString(): String

    public companion object {
        public fun now(): Double

        public fun parse(dateString: String): Double

        public fun UTC(year: Int, month: Int): Double

        public fun UTC(year: Int, month: Int, day: Int): Double

        public fun UTC(year: Int, month: Int, day: Int, hour: Int): Double

        public fun UTC(year: Int, month: Int, day: Int, hour: Int, minute: Int): Double


    }

    public interface LocaleOptions {
        public var localeMatcher: String?

        public var timeZone: String?

        public var hour12: Boolean?

        public var formatMatcher: String?

        public var weekday: String?

        public var era: String?

        public var year: String?

        public var month: String?

        public var day: String?

        public var hour: String?

        public var minute: String?

        public var second: String?

        public var timeZoneName: String?
    }
}

public inline fun dateLocaleOptions(init: Date.LocaleOptions.() -> Unit): Date.LocaleOptions {
    val result = js("new Object()")
    init(result)
    return result
}
`startIndex > endIndex`:

```kotlin
*fun substring(startIndex: Int, endIndex: Int): String

    /**
     * Attempts to reduce storage used for this string builder.
     * If the backing storage of this string builder is larger than necessary to hold its current
     * contents, then it may be resized to become more space efficient. Calling this method may, but is not
     * required to, affect the value of the [capacity] property.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    fun trimToSize() {

    }

    /**
     * Clears the content of this string builder making it
     * empty and returns this instance.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    fun clear(): StringBuilder

    /**
     * Sets the character at
     * the specified [index] to the specified [value].
     * @throws IndexOutOfBoundsException if [startIndex] is
     * less than zero, greater than the length of this string builder, or `startIndex > endIndex`.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public expect fun set(index: Int, value: Char): StringBuilder

    /**
     * Removes the character at the specified [index] from this
     * string builder and returns this instance.
     * @throws IndexOutOfBoundsException if the `Char` at the specified [index] is part of a supplementary code
     * point, this method does not remove the entire supplementary character. When [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public expect fun deleteAt(index: Int): StringBuilder

    /**
     * Removes characters in the specified range from this string builder and returns this
     * instance. When [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public expect fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder

    /**
     * Copies characters from this string builder into the [destination] character
     * array. When [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public expect fun toCharArray(destination: CharArray?, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length): String

    /**
     * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.
     * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
     */
    @SinceKotlin("1.3")
    @ExperimentalStdlibApi
    public expect fun appendRange(value: CharArray?, startIndex: Int, endIndex: Int): StringBuilder
```
fun appendRange(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder

fun insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder

fun insertRange(index: Int, value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder

public fun append(vararg value: String?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

public fun append(vararg value: Any?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

fun insertRange(index: Int, value: String?, startIndex: Int, endIndex: Int): StringBuilder

@Supress("EXTENSION_SHADOWED_BY_MEMBER.")
@Deprecated("Use append(value: Any?) instead", ReplaceWith("append(value = obj)"), DeprecationLevel.WARNING)
@kotlin.internal.InlineOnly
public inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)

public fun buildString(builderAction: StringBuilder.() -> Unit): String =
    StringBuilder().apply(builderAction).toString()

public inline fun buildString(capacity: Int, builderAction: StringBuilder.() -> Unit): String =
    StringBuilder(capacity).apply(builderAction).toString()

fun append(vararg value: String?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

fun append(vararg value: Any?): StringBuilder {
    for (item in value)
        append(item)
    return this
}

@Suppress("EXTENSION_SHADOWED_BY_MEMBER.")
@Deprecated("Use append(value: Any?) instead", ReplaceWith("append(value = obj)"), DeprecationLevel.WARNING)
@kotlin.internal.InlineOnly
public inline fun StringBuilder.append(obj: Any?): StringBuilder = this.append(obj)

public fun buildString(builderAction: StringBuilder.() -> Unit): String =
    StringBuilder().apply(builderAction).toString()
2018 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file. * n * @n npackage kotlin.dom

```
package org.w3c.dom

public external interface ItemArrayLike<out T> {
  val length: Int
  fun item(index: Int): T?
}

/**
 * Returns the view of this `ItemArrayLike<T>` collection as `List<T>`
 * @n override val size: Int get() = this@asList.length
 * override fun get(index: Int): T = when (index) {
 * in 0..lastIndex -> this@asList.item(index)
 * else -> throw IndexOutOfBoundsException("index $index is not in range [0..$lastIndex]")
 * }
 */

package kotlin.dom

import org.w3c.dom.*

/** Removes all the children from this node. */
public fun Node.clear() {
    while (hasChildNodes()) {
        removeChild(firstChild!!)
    }
}

/**
 * Creates text node and append it to the
 * element.
 * @n * @n * @n @n fun Element.appendText(text: String): Element {
 * appendChild(ownerDocument!!.createTextNode(text))
 * return this
 * }
 */

package kotlin

// NOTE: Do not author your exceptions as they are written in this file, instead use this
template:

```
public open class MyException : Exception {
    constructor() : super()
    constructor(message: String?) : super(message)
    constructor(message: String?, cause: Throwable?) : super(message, cause)
}

@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")
public actual open class Error actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor() : this(null, null) {
        throw ClassCastException("Illegal cast: $message")
    }
    constructor(cause: Throwable?) : super(cause) {
        throw IllegalStateException("lateinit property $propertyName has not been initialized")
    }
}

@Suppress("USELESS_ELVIS_RIGHT_IS_NULL")
public actual open class Exception actual constructor(message: String?, cause: Throwable?) : Throwable(message, cause ?: null) {
    actual constructor() : this(null, null) {
        throw ClassCastException("Illegal cast: $message")
    }
    constructor(cause: Throwable?) : super(cause) {
        throw IllegalStateException("lateinit property $propertyName has not been initialized")
    }
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2321
constructor() : this(null, null) { /* Exception::class.js.asDynamic().call(this, null, null) */
    actual constructor(message: String?) : this(message, null) {
        Exception::class.js.asDynamic().call(this, message, null)
    }
    actual constructor(cause: Throwable?) : this(cause) {
        RuntimeException::class.js.asDynamic().call(this, cause)
    }
}

public actual open class RuntimeException {
    constructor(message: String?, cause: Throwable?) : super(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : this()
}

public actual open class IllegalArgumentException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

public actual open class IllegalStateException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

public actual open class IndexOutOfBoundsException {
    constructor(message: String?) : RuntimeException(message)
    constructor() : super()
}

public actual open class ConcurrentModificationException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

public actual open class UnsupportedOperationException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

public actual open class NumberFormatException {
    constructor(message: String?) : IllegalArgumentException(message)
    constructor() : super()
}

public actual open class NullPointerException {
    constructor(message: String?) : RuntimeException(message)
    constructor() : super()
}

public actual open class ClassCastException {
    constructor(message: String?) : RuntimeException(message)
    constructor() : super()
}

private constructor(message: String?, cause: Throwable?) : Error(message, cause)

public actual open class NoWhenBranchMatchedException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

public actual open class UninitializedPropertyAccessException {
    constructor(message: String?, cause: Throwable?) : RuntimeException(message, cause)
    constructor(cause: Throwable?) : super(cause)
    constructor() : super()
}

package kotlin.collections

/**
 * Groups elements from the 
 [Grouping] source by key and counts elements in each group.
 * @return a 
 [Map] associating the key of each group with the count of element in the group.
 */
@SinceKotlin("1.1") fun <T, K> Grouping<T, K>.eachCount(): Map<K, Int> =
    fold(0) { acc, _ -> acc + 1 }

@media
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/
@license
package

kotlin.collections

/**
 * Groups elements from the 
 [Grouping] source by key and sums values provided by
 * the 
 [valueSelector] function for elements in each group.
 * @return a 
 [Map] associating the key of each group with the sum of values provided by
 * the 
 [valueSelector] for elements in each group.
 */
@SinceKotlin("1.1") fun <T, K> Grouping<T, K>.eachSumOf(valueSelector: (T) -> Int): Map<K, Int> =
    fold(0) { acc, e -> acc + valueSelector(e) }

/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */
*\n@file:kotlin.jvm.JvmName("GroupingKt")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin.collections

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each
 * element to get its key.
 * A [Grouping] structure serves as an intermediate step in group-and-fold operations:
 * they group elements by their keys and then fold each group with some aggregating operation.
 * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
 * To get an instance of [Grouping] use one of `groupingBy` extension functions:
 * - [Iterable.groupingBy]
 * - [Sequence.groupingBy]
 * - [Array.groupingBy]
 */

@file:kotlin.jvm.JvmName("GroupingKt")
@file:kotlin.jvm.JvmMultifileClass

package kotlin.collections

/**
 * Represents a source of elements with a [keyOf] function, which can be applied to each
 * element to get its key.
 * A [Grouping] structure serves as an intermediate step in group-and-fold operations:
 * they group elements by their keys and then fold each group with some aggregating operation.
 * It is created by attaching `keySelector: (T) -> K` function to a source of elements.
 * To get an instance of [Grouping] use one of `groupingBy` extension functions:
 * - [Iterable.groupingBy]
 * - [Sequence.groupingBy]
 * - [Array.groupingBy]
 */

@SinceKotlin("1.1")
public interface Grouping<T, out K> {
    /** Returns an [Iterator] over the elements of the source of this grouping. */
    fun sourceIterator(): Iterator<T>
    /** Extracts the key of an [element]. */
    fun keyOf(element: T): K
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group
 * sequentially, passing the previously accumulated value and the current element as arguments, and stores the
 * results in a new map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param operation function is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 * - `first`: indicates whether it's the first `element` encountered in the group.
 * @return a [Map] associating the key of each group with the result of aggregation of the group elements.
 */
@SinceKotlin("1.1")
public inline fun <T, K, R> Grouping<T, K>.aggregate(
    operation: (key: K, accumulator: R?, element: T, first: Boolean) -> R
): Map<K, R> {
    for (e in this.sourceIterator()) {
        val key = keyOf(e)
        val accumulator = operation(key, accumulator, e, accumulator == null && !destination.containsKey(key))
        destination[key] = accumulator
    }
    return destination
}

/**
 * Groups elements from the [Grouping] source by key and applies [operation] to the elements of each group sequentially, passing the previously accumulated value and the current element as arguments, and stores the results in the given destination map.
 * The key for each element is provided by the [Grouping.keyOf] function.
 * @param initialValueSelector a function that provides an initial value of accumulator for each group:
 * - `key`: the key of the group this element belongs to;
 * - `element`: the first element being encountered in that group.
 * @param operation a function that is invoked on each element with the following parameters:
 * - `key`: the key of the group this element belongs to;
 * - `accumulator`: the current value of the accumulator of the group, can be `null` if it's the first `element` encountered in the group;
 * - `first`: indicates whether it's the first `element` encountered in the group.
 * @return a [Map] associating the key of each group with the result of aggregating the group elements.
 */
@SinceKotlin("1.1")
public inline fun <T, K, R> Grouping<T, K>.fold(
    initialValueSelector: (key: K, element: T) -> R,
    operation: (key: K, accumulator: R, element: T) -> R
): Map<K, R> =
    aggregate { key, acc, e, first -> operation(key, if (first)
@Suppress("UNCHECKED_CAST")
if (first) e else operation(key, acc as S, e)
}

/**
 * Groups
 * elements from the [Grouping] source by key and applies the reducing [operation] to the elements of each group
 * sequentially starting from the second element of the group, storing the results in the given [destination] map.
 * An initial value of accumulator is the first element of the group.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the accumulator for that group and the first element of that group is also subject to the [operation].
 * @param operation a function that is invoked on each subsequent element of the group with the following parameters:
 * - `key`: the element from the source being folded;
 * - `accumulator`: the current value of the accumulator of the group;
 * - `element`: the element from the source being folded;
 * @return the [destination] map associating the key of each group with the result of accumulating the group elements.
 */
@SinceKotlin("1.1")
public inline fun <S, T : S, K, M : MutableMap<in K, S>> Grouping<T, K>.reduceTo(
    destination: M,
    operation: (key: K, accumulator: S, element: T) -> S
): M =
    aggregateTo(destination) { key, acc, e, first ->
        if (first) e else operation(key, acc as S, e)
    }

/**
 * Groups
 * elements from the [Grouping] source by key and counts elements in each group to the given [destination] map.
 * If the [destination] map already has a value corresponding to the key of some group, that value is used as an initial value of the counter for that group.
 * An initial value of counter is the first element of the group.
 * @return the [destination] map associating the key of each group with the count of elements in the group.
 */
@SinceKotlin("1.1")
public fun <T, K, M : MutableMap<in K, Int>> Grouping<T, K>.eachCountTo(destination: M): M =
    foldTo(destination, 0) { acc, _ -> acc + 1 }

@PathInfo("ioX-FogDirector-AC18")
@SinceKotlin("1.1")
public fun <T, K, M : MutableMap<in K, Double>> Grouping<T, K>.sumEachByDoubleTo(
    destination: M,
    valueSelector: (T) -> Double
): M =
    foldTo(destination, 0.0) { acc, e -> acc + valueSelector(e)}
DeprecationLevel.ERROR)
public fun atan(value: Double): Double
    @Deprecated("Use kotlin.math.atan2 instead.",
    ReplaceWith("atan2(y, x)\n\", \"kotlin.math.atan2\")
    , level = DeprecationLevel.ERROR)
public fun atan2(y: Double, x: Double):
    Double
    @Deprecated("Use kotlin.math.sin instead.",
    ReplaceWith("sin(value)\n\", \"kotlin.math.sin\")
    , level = DeprecationLevel.ERROR)
public fun sin(value: Double):
    Double
    @Deprecated("Use kotlin.math.cos instead.",
    ReplaceWith("cos(value)\n\", \"kotlin.math.cos\")
    , level = DeprecationLevel.ERROR)
public fun cos(value: Double):
    Double
    @Deprecated("Use kotlin.math.exp instead.",
    ReplaceWith("exp(value)\n\", \"kotlin.math.exp\")
    , level = DeprecationLevel.ERROR)
public fun exp(value: Double):
    Double
    @Deprecated("Use kotlin.math.sqrt instead.",
    ReplaceWith("sqrt(value)\n\", \"kotlin.math.sqrt\")
    , level = DeprecationLevel.ERROR)
public fun sqrt(value: Double):
    Double
    @Deprecated("Use kotlin.math.tan instead.",
    ReplaceWith("tan(value)\n\", \"kotlin.math.tan\")
    , level = DeprecationLevel.ERROR)
public fun tan(value: Double):
    Double
    @Deprecated("Use kotlin.math.ln instead.",
    ReplaceWith("ln(value)\n\", \"kotlin.math.ln\")
    , level = DeprecationLevel.ERROR)
public fun log(value: Double):
    Double
    @Deprecated("Use kotlin.math.pow instead.",
    ReplaceWith("pow\n\", \"kotlin.math.pow\")
    , level = DeprecationLevel.ERROR)
public fun pow(base: Double, exp: Double):
    Double
    @Deprecated("Use kotlin.math.round instead.",
    ReplaceWith("round\n\", \"kotlin.math.round\")
    , level = DeprecationLevel.ERROR)
public fun round(value: Number):
    Double
    @Deprecated("Use kotlin.math.floor instead.",
    ReplaceWith("floor\n\", \"kotlin.math.floor\")
    , level = DeprecationLevel.ERROR)
public fun floor(value: Number):
    Double
    @Deprecated("Use kotlin.math.ceil instead.",
    ReplaceWith("ceil\n\", \"kotlin.math.ceil\")
    , level = DeprecationLevel.ERROR)
public fun ceil(value: Number):
    Double
    @Deprecated("Use kotlin.math.trunc instead.",
    ReplaceWith("trunc\n\", \"kotlin.math.trunc\")
    , level = DeprecationLevel.ERROR)
public fun trunc(value: Number):
    Double
    @PublishedApi
    internal fun sinh(value: Double):
    Double
    @PublishedApi
    internal fun cosh(value: Double):
    Double
    @PublishedApi
    internal fun tanh(value: Double):
    Double
    @PublishedApi
    internal fun asinh(value: Double):
    Double
    @PublishedApi
    internal fun acosh(value: Double):
    Double
    @PublishedApi
    internal fun atanh(value: Double):
    Double
    @PublishedApi
    internal fun hypot(x: Double, y: Double):
    Double
    @PublishedApi
    internal fun expm1(value: Double):
    Double
    @PublishedApi
    internal fun log10(value: Double):
    Double
    @PublishedApi
    internal fun log2(value: Double):
    Double
    @PublishedApi
    internal fun log1p(value: Double):
    Double
    @PublishedApi
    internal fun clz32(value: Int):
    Int
    @PublishedApi
    internal fun sign(value: Number):
    Double
    @PublishedApi
    internal fun minOf(a: Int, b: Int):
    Int
    @PublishedApi
    internal fun maxOf(a: Int, b: Int):
    Int
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun minOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun maxOf(a: Double, b: Double):
    Double
    @PublishedApi
    internal fun minOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun maxOf(a: Long, b: Long):
    Long
    @PublishedApi
    internal fun minOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal fun maxOf(a: Float, b: Float):
    Float
    @PublishedApi
    internal func
```kotlin
/**
 * Calls the function will be generated:
 * `j["prop"] + j[p]` will be generated:
 * `j["prop"] = 1;` will be generated:
 * `"prop"` will be generated:
 * function test(j, p) {
 *  return
 *  j["prop"] + j[p];
 *}
 */

operator fun get(propertyName: String): Any?

/**
 * Calls the function will be generated:
 * will be translated to an assignment of [value] to the receiver indexed (with square brackets/index operation) with [propertyName].
 * E.g. for the following code:
 * function test(j, p, String, newValue: Any) {
 *  return
 *  j["prop"] = newValue;
 *  j[p] = newValue;
 *}
 */

operator fun set(propertyName: String, value: Any?): Unit

/**
 * Returns the original receiver.
 * `Json.add(other: Json)`: Js
 * `function test(j, p, value: Any) {
 *  j["prop"] = value;
 *  j[p] = value;
 *}
 */

public fun json(vararg pairs: Pair<String, Any?>): Json {
    val res: dynamic = js("({})")
    for ((name, value) in pairs) {
        res[name] = value
    }
    return res
}

/**
 * Adds key-value pairs from [other] to [this].
 * Returns the original receiver.
 */
public fun Json.add(other: Json): Json {
    val keys: Array<String> = js("Object").keys(other)
    for (key in keys) {
        if (other.asDynamic().hasOwnProperty(key)) {
            this[key] = other[key]
        }
    }
    return this
}

/**
 * `Json.stringify(o)`:
 * `function test(j, p, String, newValue: Any) {
 *  return
 *  j["prop"] = newValue;
 *  j[p] = newValue;
 *}
 */

public external object JSON {
    public fun stringify(o: Any?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?): String
    public fun stringify(o: Any?, replacer: ((key: String, value: Any?) -> Any?)?: String
    public fun stringify(o: Any?, replacer: Array<String>): String
    public fun stringify(o: Any?, replacer: Array<String>, space: Int): String
    public fun stringify(o: Any?, replacer: Array<String>, space: String): String
    public fun <T> parse(text: String): T
    public fun <T> parse(text: String, reviver: ((key: String, value: Any?) -> Any?)?: T
    public fun <T> parse(text: String, reviver: ((key: String, value: Any?)?: T
    public fun <T> parse(text: String, reviver: ((key: String, value: Any?)?: T
}
```

---

```
// region
================ Double Math ========================================

/** Computes the sine of the angle [x] given in radians.
 * Special cases:
 * - `sin(NaN)+Inf-Inf)` is `NaN`  
 */
@SinceKotlin("1.2") public actual inline fun sin(x: Double): Double = nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians.
 * Special cases:
 * - `cos(NaN)+Inf-Inf)` is `NaN`
 */
@SinceKotlin("1.2") public actual inline fun cos(x: Double): Double = nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians.
 * Special cases:
 * - `tan(NaN)+Inf-Inf)` is `NaN`
 */
@SinceKotlin("1.2") public actual inline fun tan(x: Double): Double = nativeMath.tan(x)

/**
 * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond
 * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];
 * when `abs(x) > 1` or `x = NaN` or `x = NaN` or `x = NaN` or `x = NaN` or `x = NaN`
 */
@SinceKotlin("1.2") public actual inline fun atan(x: Double): Double = nativeMath.atan(x)
```

---

```
import kotlin.internal.InlineOnly
import kotlin.js.Math as nativeMath

// region
================ Double Math ========================================

/** Computes the sine of the angle [x] given in radians.
 * Special cases:
 * 
 */
@SinceKotlin("1.2") public actual inline fun sin(x: Double): Double = nativeMath.sin(x)

/** Computes the cosine of the angle [x] given in radians.
 * Special cases:
 * 
 */
@SinceKotlin("1.2") public actual inline fun cos(x: Double): Double = nativeMath.cos(x)

/** Computes the tangent of the angle [x] given in radians.
 * Special cases:
 * 
 */
@SinceKotlin("1.2") public actual inline fun tan(x: Double): Double = nativeMath.tan(x)

/**
 * Returns the angle `theta` of the polar coordinates `(r, theta)` that correspond
 * to the rectangular coordinates `(x, y)` by computing the arc tangent of the value [y] / [x];
 * when `abs(x) > 1` or `x = NaN` or `x = NaN` or `x = NaN` or `x = NaN` or `x = NaN`
 */
@SinceKotlin("1.2") public actual inline fun atan(x: Double): Double = nativeMath.atan(x)
```
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atan2(y: Double, x: Double): Double =
nativeMath.atan2(y, x)

/**
 * Computes the hyperbolic sine of the value [x].
 * Special cases:
 *   - `sinh(NaN)` is `NaN`
 *   - `sinh(+Inf)` is `+Inf`
 *   - `sinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sinh(x: Double): Double =
nativeMath.sinh(x)

/**
 * Computes the hyperbolic cosine of the value [x].
 * Special cases:
 *   - `cosh(NaN)` is `NaN`
 *   - `cosh(+Inf)` is `+Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cosh(x: Double): Double =
nativeMath.cosh(x)

/**
 * Computes the hyperbolic tangent of the value [x].
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(+Inf)` is `1.0`
 *   - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tanh(x: Double): Double =
nativeMath.tanh(x)

/**
 * Computes the inverse hyperbolic sine of the value [x].
 * Special cases:
 *   - `asinh(NaN)` is `NaN`
 *   - `asinh(+Inf)` is `+Inf`
 *   - `asinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asinh(x: Double): Double =
nativeMath.asinh(x)

/**
 * Computes the inverse hyperbolic cosine of the value [x].
 * Special cases:
 *   - `acosh(NaN)` is `NaN`
 *   - `acosh(x)` is `NaN` when `x < 1`
 *   - `acosh(+Inf)` is `+Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun acosh(x: Double): Double =
nativeMath.acosh(x)

/**
 * Computes the inverse hyperbolic tangent of the value [x].
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(x)` is `NaN` when `x > 1` or `x < -1`
 *   - `tanh(1.0)` is `+Inf`
 *   - `tanh(-1.0)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atanh(x: Double): Double =
nativeMath.atanh(x)

/**
 * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.
 * Special cases:
 *   - returns `+Inf` if any of arguments is infinite
 *   - returns `NaN` if any of arguments is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun hypot(x: Double, y: Double): Double =
nativeMath.hypot(x, y)

/**
 * Computes the positive square root of the value [x].
 * Special cases:
 *   - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sqrt(x: Double): Double =
nativeMath.sqrt(x)

/**
 * Computes Euler's number `e` raised to the power of the value [x].
 * Special cases:
 *   - `exp(NaN)` is `NaN`
 *   - `exp(+Inf)` is `+Inf`
 *   - `exp(-Inf)` is `0.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun exp(x: Double): Double =
nativeMath.exp(x)

/**
 * Computes `exp(x) - 1`.
 * Special cases:
 *   - `expm1(NaN)` is `NaN`
 *   - `expm1(+Inf)` is `+Inf`
 *   - `expm1(-Inf)` is `-1.0`
 * @see [exp] function
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun expm1(x: Double): Double =
nativeMath.expm1(x)

/**
 * Computes the natural logarithm (base `E`) of the value [x].
 * Special cases:
 *   - `ln(x)` is `NaN` when `x < 0`
 *   - `ln(+Inf)` is `+Inf`
 *   - `ln(0.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public actual fun log(x: Double, base: Double) =
    if (base <= 0.0 || base == 1.0) return Double.NaN
    return nativeMath.log(x) / nativeMath.log(base)

/**
 * Computes the common logarithm (base 10) of the value [x].
 * Special cases:
 *   - `log10(x)` is `NaN` when `x < 0`
 *   - `log10(+Inf)` is `+Inf`
 *   - `log10(0.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public actual inline fun log10(x: Double): Double =
nativeMath.log10(x)

/**
 * Computes the binary logarithm (base 2) of the value [x].
 * Special cases:
 *   - `log2(x)` is `NaN` when `x < 0`
 *   - `log2(+Inf)` is `+Inf`
 *   - `log2(0.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public actual inline fun log2(x: Double): Double =
nativeMath.log2(x)

/**
 * Computes `ln1p(x)`.
 * Special cases:
 *   - `ln1p(NaN)` is `NaN`
 *   - `ln1p(+Inf)` is `+Inf`
 *   - `ln1p(-1.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public actual inline fun ln1p(x: Double): Double =
nativeMath.log1p(x)
nativeMath.log1p(x)\n\n**\n
* Rounds the given value \([x]\) to an integer towards positive infinity.\n\n* @return the smallest double value that is greater than or equal to the given value \([x]\) and is a mathematical integer.\n\n* @special cases:\n* - `ceil(x)` is \(\`x`\) where \(\`x`\) is \`NaN` or \`+Inf` or \`-Inf` or already a mathematical integer.\n\n* @since Kotlin("1.2")\n\n* public actual inline fun ceil(x: Double): Double = \nnativeMath.ceil(x).unsafeCast<Double>() // TODO: Remove unsafe cast after removing public js.math\n\n**\n
* Rounds the given value \([x]\) to an integer towards negative infinity.\n\n* @return the largest double value that is smaller than or equal to the given value \([x]\) and is a mathematical integer.\n\n* @special cases:\n* - `floor(x)` is \(\`x`\) where \(\`x`\) is \`NaN` or \`+Inf` or \`-Inf` or already a mathematical integer.\n\n* @since Kotlin("1.2")\n\n* public inline fun floor(x: Double): Double = \nnativeMath.floor(x).unsafeCast<Double>()

**\n
* Rounds the closest integer with ties rounded towards even integer.\n\n* @special cases:\n* - `truncate(x)` is \(\`x`\) where \(\`x`\) is \`NaN` or \`+Inf` or \`-Inf` or already a mathematical integer.\n\n* @since Kotlin("1.2")\n\n* public actual fun truncate(x: Double): Double = nativeMath.trunc(x)

**\n
* Rounds the closest integer with ties rounded towards even integer.\n\n* @special cases:\n* - `round(x)` is \(\`x`\) where \(\`x`\) is \`NaN` or \`+Inf` or \`-Inf` or already a mathematical integer.\n\n* @since Kotlin("1.2")\n\n* public fun round(x: Double): Double {
    if (x % 0.5 != 0.0) {
        return nativeMath.round(x).unsafeCast<Double>()
    }
    val floor = floor(x)
    return if (floor % 2 == 0.0) floor else ceil(x)
}

**\n
* Returns the absolute value of the given value \([x]\).\n\n* @special cases:\n* - `abs(NaN)` is `NaN`\n\n* @see absoluteValue extension property\n\n**\n
* Returns the sign of the given value \([x]\):\n*   - `-1.0` if the value is negative,\n*   - `0.0` if the value is zero,\n*   - `1.0` if the value is positive\n\n* @special case:\n*   - `sign(NaN)` is `NaN`\n\n**\n
* Returns the smaller of two values.\n\n* @special cases:\n* - `b.pow(0.0)` is `1.0`\n* - `b.pow(1.0) == b`\n* - `b.pow(NaN)` is `NaN`\n* - `NaN.pow(x)` is `NaN`\n* - `0.0.pow(x)` is `NaN`\n* - `x != 0.0`\n* - `b.pow(Inf)` is `NaN`\n* - `abs(b) == 1.0`\n* - `b.pow(x)` is `NaN`\n* - `b < 0` and `x` is finite and not an integer\n\n* @since Kotlin("1.2")\n\n* public actual inline fun Double.min(a: Double, b: Double): Double = \nnativeMath.min(a, b).unsafeCast<Double>()

**\n
* Returns the ulp (unit in the last place) of this value.\n\n* @special cases:\n* - `NaN.ulp` is `NaN`\n* - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`\n* - `0.0.ulp` is `Double.MIN_VALUE`\n* - `abs(b) == 1.0`\n* - `b < 0` and `x` is finite and not an integer\n* - `x != 0.0`\n* - `b.pow(Inf)` is `NaN`\n* - `abs(b) == 1.0`\n* - `b.pow(x)` is `NaN`\n* - `b < 0` and `x` is finite and not an integer\n\n* @since Kotlin("1.2")\n\n* public actual val Double.ulp: Double get() = \nwhen {\n    this < 0 -> (-this).ulp\n    this.isNaN() || this == Double.POSITIVE_INFINITY \n    this == Double.MAX_VALUE\n    this.nextDown() \n    else -> this.nextUp() \n    else -> Double.MIN_VALUE
}

**\n
* Returns the ulp (unit in the last place) of this value.\n\n* @special cases:\n* - `NaN.ulp` is `NaN`\n* - `x.ulp` is `+Inf` when `x` is `+Inf` or `-Inf`\n* - `0.0.ulp` is `Double.MIN_VALUE`\n\n* @since Kotlin("1.2")\n\n* public actual val Double.ulp: Double get() = \nwhen {\n    this < 0 -> (-this).ulp\n    this.isNaN() \n    this == Double.POSITIVE_INFINITY \n    this == Double.MAX_VALUE \n    this.nextDown() \n    else -> this.nextUp() \n    else -> Double.MIN_VALUE
}
Returns the [Double] value nearest to this value in direction of negative infinity.

```kotlin
@SinceKotlin("1.2")
public actual fun Double.nextDown(): Double = when {
    this.isNaN() || this == Double.NEGATIVE_INFINITY -> this
    this == 0.0 -> -Double.MIN_VALUE
    else ->
        Double.fromBits(this.toRawBits() + if (this > 0) -1 else 1) // Rounds this [Double] value to the nearest integer and converts the result to [Int].
}
```

### nextTowards

```
/**
 * Returns the [Double] value nearest to this value from this value towards the value [to].
 *
 * Special cases:
 *   - `Double.NaN.nextTowards(y)` returns `NaN`
 *   - `Double.NEGATIVE_INFINITY.nextTowards(y)` returns `y`
 *   - `Double.POSITIVE_INFINITY.nextTowards(y)` returns `y`
 *   - `Double.ZERO.nextTowards(y)` returns `y`
 *
 * @since 1.2
 */
@SinceKotlin("1.2")
public actual fun Double.nextTowards(to: Double): Double = when {
    this.isNaN() || to.isNaN() -> Double.NaN
    to == this -> to
    to > this -> this.nextUp()
    else /* to < this */ -> this.nextDown()
}
```

### roundToInt

```
/**
 * Rounds this [Double] value to the nearest integer and converts the result to [Int].
 * Ties are rounded towards positive infinity.
 *
 * Special cases:
 *   - `Double.MAX_VALUE.roundToInt()` returns `Int.MAX_VALUE`
 *   - `Double.MIN_VALUE.roundToInt()` returns `Int.MIN_VALUE`
 *
 * @since 1.2
 */
@SinceKotlin("1.2")
public actual fun Double.roundToInt(): Int = when {
    this.isNaN() -> throw IllegalArgumentException("Cannot round NaN value.")
    this > Int.MAX_VALUE -> Int.MAX_VALUE
    this < Int.MIN_VALUE -> Int.MIN_VALUE
    else ->
        nativeMath.round(this).unsafeCast<Double>().toInt()
}
```

### roundToLong

```
/**
 * Rounds this [Double] value to the nearest integer and converts the result to [Long].
 * Ties are rounded towards positive infinity.
 *
 * Special cases:
 *   - `Double.MAX_VALUE.roundToLong()` returns `Long.MAX_VALUE`
 *   - `Double.MIN_VALUE.roundToLong()` returns `Long.MIN_VALUE`
 *
 * @since 1.2
 */
@SinceKotlin("1.2")
public actual fun Double.roundToLong(): Long = when {
    this.isNaN() -> throw IllegalArgumentException("Cannot round NaN value.")
    this > Long.MAX_VALUE -> Long.MAX_VALUE
    this < Long.MIN_VALUE -> Long.MIN_VALUE
    else ->
        nativeMath.round(this).unsafeCast<Double>().toLong()
}
```

### Float Math

```
/** Computes the sine of the angle [x] given in radians.
 *
 * Special cases:
 *   - `sin(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun sin(x: Float): Float = nativeMath.sin(x.toDouble()).toFloat()

/** Computes the cosine of the angle [x] given in radians.
 *
 * Special cases:
 *   - `cos(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun cos(x: Float): Float = nativeMath.cos(x.toDouble()).toFloat()

/** Computes the tangent of the angle [x] given in radians.
 *
 * Special cases:
 *   - `tan(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun tan(x: Float): Float = nativeMath.tan(x.toDouble()).toFloat()

/** Computes the arc sine of the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 *
 * Special cases:
 *   - `asin(0)` returns `0`
 *   - `asin(1)` returns `PI/2`
 *   - `asin(-1)` returns `-PI/2`
 *   - `asin(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun asin(x: Float): Float = nativeMath.asin(x.toDouble()).toFloat()

/** Computes the arc cosine of the value [x]; the returned value is an angle in the range from `0.0` to `PI` radians.
 *
 * Special cases:
 *   - `acos(0)` returns `PI/2`
 *   - `acos(1)` returns `0.0`
 *   - `acos(-1)` returns `PI`
 *   - `acos(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun acos(x: Float): Float = nativeMath.acos(x.toDouble()).toFloat()

/** Computes the arc tangent of the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 *
 * Special cases:
 *   - `atan(0)` returns `0`
 *   - `atan(1)` returns `PI/4`
 *   - `atan(-1)` returns `-PI/4`
 *   - `atan(NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun atan(x: Float): Float = nativeMath.atan(x.toDouble()).toFloat()

/** Computes the arc tangent of the value [y] and the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 *
 * Special cases:
 *   - `atan2(0, 0)` returns `0`
 *   - `atan2(1, 0)` returns `PI/2`
 *   - `atan2(-1, 0)` returns `-PI/2`
 *   - `atan2(0, -1)` returns `-PI`
 *   - `atan2(0, 1)` returns `0`
 *   - `atan2(-1, 0)` returns `-PI/2`
 *   - `atan2(1, -1)` returns `-PI/4`
 *   - `atan2(-1, -1)` returns `-3PI/4`
 *   - `atan2(1, 1)` returns `PI/4`
 *   - `atan2(-1, 1)` returns `-3PI/4`
 *   - `atan2(NaN, x)` returns `NaN`
 *   - `atan2(y, NaN)` returns `NaN`
 */
@SinceKotlin("1.2")
public actual inline fun atan2(y: Float, x: Float): Float = nativeMath.atan2(y.toDouble(), x.toDouble()).toFloat()
```

---

**Open Source Used In IOx-FogDirector-AC18 1.17.0 2330**
```kotlin
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sinh(x: Float): Float = nativeMath.sinh(x.toDouble()).toFloat()

/**
 * Computes the hyperbolic cosine of the value [x].
 *
 * Special cases:
 *   - `cosh(NaN)` is `NaN`
 *   - `cosh(+Inf)` is `+Inf`
 *   - `cosh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun cosh(x: Float): Float = nativeMath.cosh(x.toDouble()).toFloat()

/**
 * Computes the hyperbolic tangent of the value [x].
 *
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(+Inf)` is `1.0`
 *   - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun tanh(x: Float): Float = nativeMath.tanh(x.toDouble()).toFloat()

/**
 * Computes the inverse hyperbolic sine of the value [x].
 *
 * The returned value is `y` such that `sinh(y) == x`. Special cases:
 *   - `asinh(NaN)` is `NaN`
 *   - `asinh(+Inf)` is `+Inf`
 *   - `asinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun asinh(x: Float): Float = nativeMath.asinh(x.toDouble()).toFloat()

/**
 * Computes the inverse hyperbolic cosine of the value [x].
 *
 * The returned value is positive `y` such that `cosh(y) == x`. Special cases:
 *   - `acosh(NaN)` is `NaN`
 *   - `acosh(x)` is `NaN` when `x < 1`
 *   - `acosh(+Inf)` is `+Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun acosh(x: Float): Float = nativeMath.acosh(x.toDouble()).toFloat()

/**
 * Computes the inverse hyperbolic tangent of the value [x].
 *
 * The returned value is `y` such that `tanh(y) == x`. Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(x)` is `NaN` when `x > 1` or `x < -1`
 *   - `tanh(1.0)` is `+Inf`
 *   - `tanh(-1.0)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun atanh(x: Float): Float = nativeMath.atanh(x.toDouble()).toFloat()

/**
 * Computes `sqrt(x^2 + y^2)` without intermediate overflow or underflow.
 *
 * Special cases:
 *   - returns `+Inf` if any of arguments is infinite
 *   - returns `NaN` if any of arguments is `NaN`
 *   - returns `NaN` when `x < 0` or `y < 0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun hypot(x: Float, y: Float): Float = nativeMath.hypot(x.toDouble(), y.toDouble()).toFloat()

/**
 * Computes the positive square root of the value [x].
 *
 * Special cases:
 *   - `sqrt(x)` is `NaN` when `x < 0` or `x` is `NaN`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun sqrt(x: Float): Float = nativeMath.sqrt(x.toDouble()).toFloat()

/**
 * Computes Euler's number `e` raised to the power of the value [x].
 *
 * Special cases:
 *   - `exp(NaN)` is `NaN`
 *   - `exp(+Inf)` is `+Inf`
 *   - `exp(-Inf)` is `0.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun exp(x: Float): Float = nativeMath.exp(x.toDouble()).toFloat()

/**
 * Computes `exp(x) - 1`.
 *
 * This function can be implemented to produce more precise result for [x] near zero.
 *
 * Special cases:
 *   - `expm1(NaN)` is `NaN`
 *   - `expm1(x)` is `NaN` when `x > 1` or `x < -1`
 *   - `expm1(+Inf)` is `+Inf`
 *   - `expm1(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun expm1(x: Float): Float = nativeMath.expm1(x.toDouble()).toFloat()

/**
 * Computes the logarithm of the value [x] to the given [base].
 *
 * Special cases:
 *   - `log(x, b)` is `NaN` if either `x` or `b` are `NaN`
 *   - `log(x, b)` is `NaN` when `x < 0` or `b <= 0`
 *   - `log(x, b)` is `-Inf` for `b > 1` and `-Inf` for `b < 1`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun log(x: Float, base: Float): Float = nativeMath.log(x.toDouble(), base.toDouble()).toFloat()

/**
 * Computes `ln1p(x)`.
 *
 * This function can be implemented to produce more precise result for [x] near zero.
 *
 * Special cases:
 *   - `ln1p(NaN)` is `NaN`
 *   - `ln1p(x)` is `NaN` when `x < -1.0`
 *   - `ln1p(+Inf)` is `+Inf`
 *   - `ln1p(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline fun ln1p(x: Float): Float = nativeMath.ln1p(x.toDouble()).toFloat()
```
@see [expm1] function

* Rounds the given value [x] to an integer towards positive

* Rounds the given value [x] to an integer towards 

* @return the largest Float value that is smaller than or equal to the given value [x] and is a 

* Rounds the given value [x] to an integer towards zero,

* Returns the absolute value of the given value [x].

* Returns the sign of the given value [x]:

* Returns the smaller of two values.

* Returns the greater of two values.

* Returns the absolute value of this value.

* Returns the sign of this value:

* Returns this value with the sign bit same as of the [sign] value.

* If [sign] is `NaN` the sign of the result is undefined.

* Rounds this value with the sign bit same as of the [sign] value.

* Converts this value to a `Int`.
toDouble().roundToLong() rounds this [Float] value to the nearest integer and converts the result to [Long].

* `Ties are rounded towards positive infinity.`

* Special cases:
  - `x.roundToLong()` is `Long.MAX_VALUE` when `x > Long.MAX_VALUE`
  - `-x.roundToLong()` is `Long.MIN_VALUE` when `-x < Long.MIN_VALUE`

* `@Throws(IllegalArgumentException when this value is NaN)`

```java
/**
 * Rounds this [Float] value to the nearest integer and converts the result to
 * [Long].
 * `Ties are rounded towards positive infinity.`
 * Special cases:
 *   - `x > Long.MAX_VALUE` returns `Long.MAX_VALUE`
 *   - `-x < Long.MIN_VALUE` returns `-Long.MIN_VALUE`
 *   - `x < Long.MIN_VALUE` and `-x > Long.MAX_VALUE` returns `NaN`
 *   - `0.0` returns `0.0` (not `Long.MIN_VALUE`)
 *   - `-0.0` returns `-0.0` (not `-Long.MIN_VALUE`)
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
@InlineOnly
public actual fun Float.roundToLong(): Long =
toDouble().roundToLong() endregion
```

**Integer Math**

```java
/**
 * Returns the absolute value of the given value [n].
 * Special cases:
 *   - `Long.MIN_VALUE.absoluteValue` is `Long.MIN_VALUE`
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
public actual fun Long.absoluteValue: Long get() = abs(this)

/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 *   - `0` if the value is zero,
 *   - `1` if the value is positive
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
public actual val Long.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}
```

```java
/**
 * Returns the absolute value of the given value [n].
 * Special cases:
 *   - `Double.MAX_VALUE` returns `Double.MAX_VALUE`
 *   - `Double.MIN_VALUE` returns `Double.MIN_VALUE`
 *   - `NaN` returns `NaN`
 *   - `-0.0` returns `-0.0` (not `-Double.MIN_VALUE`)
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
public actual fun Double.abs(n: Double): Double =

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public actual inline fun min(a: Long, b: Long): Long = if (a <= b) a else b
```

```java
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public actual inline fun max(a: Long, b: Long): Long = if (a >= b) a else b
```

```java
/**
 * Returns the absolute value of this value.
 * Special cases:
 *   - `Int.MIN_VALUE` returns `Int.MIN_VALUE` due to an overflow
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline val Int.absoluteValue: Int get() = abs(this)
```

```java
/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 *   - `0` if the value is zero,
 *   - `1` if the value is positive
 */
@SinceKotlin("1.2")
public actual val Int.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}
```

```java
/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public actual fun Int.min(a: Int, b: Int): Int =
```

```java
/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
@Suppress("NOTHING_TO_INLINE")
public actual inline fun Int.max(a: Int, b: Int): Int =
```

```java
/**
 * Returns the absolute value of this value.
 * Special cases:
 *   - `Long.MIN_VALUE` returns `Long.MIN_VALUE` due to an overflow
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.2")
@InlineOnly
public actual inline val Long.absoluteValue: Long get() = abs(this)
```

```java
/**
 * Returns the sign of this value:
 *   - `-1` if the value is negative,
 *   - `0` if the value is zero,
 *   - `1` if the value is positive
 */
@SinceKotlin("1.2")
public actual val Long.sign: Int get() = when {
    this < 0 -> -1
    this > 0 -> 1
    else -> 0
}
```

```java
/**
 * Returns the absolute value of the given value [n].
 * Special cases:
 *   - `Long.MIN_VALUE` returns `Long.MIN_VALUE` due to an overflow
 * @See: [FloatingPointMath#isFinite]
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Int.countOneBits(): Int {
    // Hacker's Delight 5-1 algorithm
    var v = this
    v = (v and 0x55555555) + (v.ushr(1) and 0x55555555)
    v = (v and 0x33333333) + (v.ushr(2) and 0x33333333)
    v = (v and 0x0F0F0F0F) + (v.ushr(4) and 0x0F0F0F0F)
    v = (v and 0x00FF00FF) + (v.ushr(8) and 0x00FF00FF)
    v = (v and 0x0000FFFF) + (v.ushr(16))
    return
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0
 Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.

 Hacker's Delight 5-4 algorithm for expressing countTrailingZeroBits with countLeadingZeroBits
 Int.SIZE_BITS - (this or -this).inv().countLeadingZeroBits()\n * Counts the number of least significant set bits of this [Int] number,\n * or zero, if this number is zero.

 Hacker's Delight 2-1 algorithm for isolating rightmost 1-bit
 this and -this\n * Rotates the binary representation of this [Int] number left by the specified [bitCount] number of bits.\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.\n * number.rotateLeft(-n) == number.rotateRight(n)\n
 Counts the number of set bits in the binary representation of this [Long] number.\n * Counts the number of least significant set bits that are zero in the binary representation of this [Long] number.\n * Returns a number having a single bit set in the position of the most significant set bit of this [Long] number,\n * or zero, if this number is zero.\n
 Returns a number having a single bit set in the position of the least significant set bit of this [Long] number,\n * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.\n * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.\n * number.rotateLeft(-n) == number.rotateRight(n)
Rotating by a multiple of \([\text{Long.SIZE_BITS}\)] (64) returns the same number, or more generally

\[*\text{number.rotateLeft}(n) == \text{number.rotateLeft}((n \% 64))]\*

```kotlin
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public actual fun Long.rotateLeft(bitCount: Int): Long {
    if ((bitCount and 31) != 0) {
        val low = this.low
        val high = this.high
        val newLow = low.shl(bitCount) or high.ushr(-bitCount)
        val newHigh = high.shl(bitCount) or low.ushr(-bitCount)
    return if ((bitCount and 32) == 0) Long(newLow, newHigh) else Long(newHigh, newLow)
    } else {
        return if ((bitCount and 32) == 0) this else Long(high, low)
    }
}

/**
 * Rotates the binary representation of this
 \[\text{[Long]}\] number right by the specified \[\text{[bitCount]}\] number of bits.\n * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.\n * \[*\text{Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count}/*\]

\[*\text{number.rotateRight}(-n) == \text{number.rotateLeft}(n)\]*

\*/
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@kotlin.internal.InlineOnly
public actual inline fun
Long.rotateRight(bitCount: Int): Long = rotateLeft(-bitCount)
```

```kotlin
// It's workaround
for KT-19672 since we can fix it properly until KT-11265 isn't fixed.\n
inline fun <T, S>
Promise<Promise<T>>.then(
    noinline onFulfilled: (T) -> S
): Promise<S> {
    return this.unsafeCast<Promise<T>>().then(onFulfilled)
}
```
reference for the given `KClass`.
*/
val <T : Any> KClass<T>.js: JsClass<T>
    get() = (this as
KClassImpl<T>).jClass

/**
 * Obtains a `KClass` instance for the given constructor reference.
 */
val <T : Any> JsClass<T>.kotlin: KClass<T>
    get() = getKClass(this)

/*
 * Copyright 2010-2018 JetBrains s.r.o.
 * Use of this source code is governed by the Apache 2.0 license
 * that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect.js

import kotlin.reflect.*

internal abstract class KClassImpl<T : Any>(
    internal open val jClass: JsClass<T>
) :
KClass<T> {
    override val annotations: List<Annotation>
        get() = TODO()
    override val constructors: Collection<KFunction<T>>
        get() = TODO()
    override val isAbstract: Boolean
        get() = TODO()
    override val isCompanion: Boolean
        get() = TODO()
    override val isData: Boolean
        get() = TODO()
    override val isOpen: Boolean
        get() = TODO()
    override val isSealed: Boolean
        get() = TODO()
    override val members: Collection<KCallable<*>>
        get() = TODO()
    override val nestedClasses:
        Collection<KClass<*>>
        get() = TODO()
    override val objectInstance: T?
        get() = TODO()
    override val qualifiedName: String?
        get() = TODO()
    override val supertypes: List<KType>
        get() = TODO()
    override val typeParameters: List<KTypeParameter>
        get() = TODO()
    override val sealedSubclasses: List<KClass<out T>>
        get() = TODO()
    override val visibility: KVisibility?
        get() = TODO()

    override fun equals(other: Any?): Boolean {
        return other is KClassImpl<*> && jClass ==
other.jClass
    }

    // TODO: use FQN
    override fun hashCode(): Int = simpleName?.hashCode() ?: 0

    override fun toString(): String {
        // TODO: use FQN
        return "class $simpleName"
    }
}

internal class SimpleKClassImpl<T : Any>(jClass: JsClass<T>) : KClassImpl<T>(jClass) {
    override val simpleName: String? = jClass.asDynamic().`$metadata$`?.simpleName.unsafeCast<String?>()

    override fun isInstance(value: Any?): Boolean {
        return jsIsType(value, jClass)
    }
}

internal class PrimitiveKClassImpl<T : Any>(
    jClass: JsClass<T>,
    private val givenSimpleName: String,
    private val isInstanceFunction: (Any?) -> Boolean
) : KClassImpl<T>(jClass) {
    override fun equals(other: Any?): Boolean {
        if (other !is PrimitiveKClassImpl<*>) return false
        return super.equals(other) && givenSimpleName == other.givenSimpleName
    }

    override val simpleName: String? get() = givenSimpleName

    override fun isInstance(value: Any?): Boolean {
        return isInstanceFunction(value)
    }
}

internal object NothingKClassImpl : KClassImpl<Nothing>(js("Object")) {
    override val simpleName: String = "Nothing"

    override fun isInstance(value: Any?): Boolean = false
    override val jsClass: JsClass<Nothing> = throw UnsupportedOperationException("There's no native JS class for Nothing type")

    override fun isInstanceFunction(value: Any?): Boolean = other == false

    override fun isSuperEquals(other: KType): Boolean {
        return other is NothingKClassImpl && other.simpleName == simpleName
    }
}

// TODO: Remove once JsReflectionAPICallChecker supports more reflection types
@file:Suppress("Unsupported")

import kotlin.reflect.*
import kotlin.reflect.js.*
import kotlin.reflect.js.internal.*

// TODO: Remove once Kotlin reflection is not supported
@file:Suppress("Unsupported")
import kotlin.reflect.*
import kotlin.reflect.js.*
import kotlin.reflect.js.internal.*

// a package is omitted to get declarations directly under the module
// TODO: Remove once
JsReflectionAPICallChecker.supports more reflection types
createInvariantKTypeProjection(type: KType): KTypeProjection =
KTypeProjection.invariant(type)
@JsName("createContravariantKTypeProjection")\ninternal fun
createContravariantKTypeProjection(type: KType): KTypeProjection =
KTypeProjection.contravariant(type)/* * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
* @package kotlin.reflect.js.internal
* @import kotlin.reflect.
* @internal class KTypeImpl(
*     override val classifier: KClassifier,
*     override val arguments: List<KTypeProjection>,
*     override val isMarkedNullable: Boolean
* ) : KType {
* override val annotations: List<Annotation>
*     get() =
*         emptyList()
* override fun equals(other: Any?): Boolean =
*         other is KTypeImpl &&
*         classifier == other.classifier &&
*         arguments == other.arguments &&
*         isMarkedNullable == other.isMarkedNullable
* override fun hashCode(): Int =
*         (classifier.hashCode() * 31 + arguments.hashCode()) * 31 +
*         isMarkedNullable.hashCode()
* override fun toString(): String {
*     val kClass = (classifier as? KClass<*>)
*     val classifierName = when {
*         kClass == null ->"
*         else -> "(non-denotable type)"
*     }\n*     val args =\n*         if (arguments.isEmpty())"
*         else arguments.joinToString(", ", "<", ">") { it.asString() }\n*     val nullable = if (isMarkedNullable) '?' else"
*     return classifierName + args + nullable\n* // TODO: this should be the implementation of KTypeProjection.toString, see KT-3007
* private fun KTypeProjection.asString(): String {
*     if (variance == null) return '*'
*     return variance.prefixString() +
*         type.toString()\n* }\n* }\n* @JsName("DynamicKType")
* internal fun KVariance.prefixString() =
*     when (this) {
*         INVAR:
*             ""
*         IN:
*             "in "
*         OUT:
*             "out "
*     }\n* }\n* @JsName("PrimitiveClasses")
* internal Object DynamicKType : KType {
*     override val classifier: KClassifier =
*         null
*     override val arguments: List<KTypeProjection> = emptyList()
*     override val isMarkedNullable: Boolean =
*         false
*     override val annotations: List<Annotation> = emptyList()
*     override fun toString(): String =
*         "dynamic"
* }\n* @JsName("KTypeParameterImpl")
* internal data class KTypeParameterImpl(
*     override val name: String,
*     override val upperBounds: List<KType>,
*     override val variance: KVariance,
*     override val isReified: Boolean
* ) : KTypeParameter {
*     override fun toString(): String = name
* }\n* @JsName("@IsName("PrimitiveClasses")\n*     internal object PrimitiveClasses {\n*         @JsName("anyClass")\n*             val anyClass = PrimitiveKClassImpl(js("Object"), unsafeCast<JsClass<Any>>(),
*                 "{ it is Any "}\n*         @JsName("numberClass")\n*             val numberClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Number>>(),
*                     "{ it is Number "})\n*         @JsName("nothingClass")\n*             val nothingClass = NothingKClassImpl\n*         @JsName("booleanClass")\n*             val booleanClass =
*                 PrimitiveKClassImpl(js("Boolean"), unsafeCast<JsClass<Boolean>>(),
*                     "{ it is Boolean "})\n*         @JsName("byteClass")\n*             val byteClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Byte>>(),
*                     "{ it is Byte "})\n*         @JsName("shortClass")\n*             val shortClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Short>>(),
*                     "{ it is Short "})\n*         @JsName("intClass")\n*             val intClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Int>>(),
*                     "{ it is Int "})\n*         @JsName("floatClass")\n*             val floatClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Float>>(),
*                     "{ it is Float "})\n*         @JsName("doubleClass")\n*             val doubleClass =
*                 PrimitiveKClassImpl(js("Number"), unsafeCast<JsClass<Double>>(),
*                     "{ it is Double "})\n*         @JsName("arrayClass")\n*             val arrayClass =
*                 PrimitiveKClassImpl(js("Array"), unsafeCast<JsClass<Array<*>>>(),
*                     "{ it is Array<*> "})\n*         @JsName("stringClass")\n*             val stringClass =
*                 PrimitiveKClassImpl(js("String"), unsafeCast<JsClass<String>>(),
*                     "{ it is String "})\n*         @JsName("throwableClass")\n*             val throwableClass =
PrimitiveKClassImpl(js(\"Error\").unsafeCast<JsClass<Throwable>>(), \"Throwable\", { it is Throwable })\n\n
@JsName(\"booleanArrayClass\")\n val booleanArrayClass =
PrimitiveKClassImpl(js(\"Array\").unsafeCast<JsClass<BooleanArray>>(), \"BooleanArray\", { it is BooleanArray
})\n\n @JsName(\"charArrayClass\")\n val charArrayClass =
PrimitiveKClassImpl(js(\"Uint16Array\").unsafeCast<JsClass<CharArray>>(), \"CharArray\", { it is CharArray
})\n\n @JsName(\"byteArrayClass\")\n val byteArrayClass =
PrimitiveKClassImpl(js(\"Int8Array\").unsafeCast<JsClass<ByteArray>>(), \"ByteArray\", { it is ByteArray })\n\n
@JsName(\"shortArrayClass\")\n val shortArrayClass =
PrimitiveKClassImpl(js(\"Int16Array\").unsafeCast<JsClass<ShortArray>>(), \"ShortArray\", { it is ShortArray
})\n\n @JsName(\"intArrayClass\")\n val intArrayClass =
PrimitiveKClassImpl(js(\"Int32Array\").unsafeCast<JsClass<IntArray>>(), \"IntArray\", { it is IntArray })\n\n
@JsName(\"longArrayClass\")\n val longArrayClass =
PrimitiveKClassImpl(js(\"Array\").unsafeCast<JsClass<LongArray>>(), \"LongArray\", { it is LongArray })\n\n
@JsName(\"floatArrayClass\")\n val floatArrayClass =
PrimitiveKClassImpl(js(\"Float32Array\").unsafeCast<JsClass<FloatArray>>(), \"FloatArray\", { it is FloatArray
})\n\n @JsName(\"doubleArrayClass\")\n val doubleArrayClass =
PrimitiveKClassImpl(js(\"Float64Array\").unsafeCast<JsClass<DoubleArray>>(), \"DoubleArray\", { it is
DoubleArray })\n\n @JsName(\"functionClass\")\n fun functionClass(arity: Int): KClassImpl<Any> {\n
return functionClasses.get(arity) ?: run {\n
val result =
PrimitiveKClassImpl(js(\"Function\").unsafeCast<JsClass<Any>>(), \"Function$arity\",\n
{ jsTypeOf(it) === \"function\" && it.asDynamic().length === arity })\n
functionClasses.asDynamic()[arity]
= result\n
result\n
}\n }\n}\n\nprivate val functionClasses =
arrayOfNulls<KClassImpl<Any>>(0)","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n */\n\n// a package is omitted to get declarations directly under the module\n\nimport
kotlin.reflect.*\nimport kotlin.reflect.js.internal.*\n\n@JsName(\"getKClass\")\ninternal fun <T : Any>
getKClass(jClass: JsClass<T>): KClass<T> =
getOrCreateKClass(jClass)\n\n@JsName(\"getKClassFromExpression\")\ninternal fun <T : Any>
getKClassFromExpression(e: T): KClass<T> =\n when (jsTypeOf(e)) {\n
\"string\" ->
PrimitiveClasses.stringClass\n
\"number\" -> if (jsBitwiseOr(e, 0).asDynamic() === e)
PrimitiveClasses.intClass else PrimitiveClasses.doubleClass\n
\"boolean\" -> PrimitiveClasses.booleanClass\n
\"function\" -> PrimitiveClasses.functionClass(e.asDynamic().length)\n
else -> {\n
when {\n
e
is BooleanArray -> PrimitiveClasses.booleanArrayClass\n
e is CharArray ->
PrimitiveClasses.charArrayClass\n
e is ByteArray -> PrimitiveClasses.byteArrayClass\n
e is
ShortArray -> PrimitiveClasses.shortArrayClass\n
e is IntArray -> PrimitiveClasses.intArrayClass\n
e is LongArray -> PrimitiveClasses.longArrayClass\n
e is FloatArray ->
PrimitiveClasses.floatArrayClass\n
e is DoubleArray -> PrimitiveClasses.doubleArrayClass\n
e is
KClass<*> -> KClass::class\n
e is Array<*> -> PrimitiveClasses.arrayClass\n
else -> {\n
val constructor = js(\"Object\").getPrototypeOf(e).constructor\n
when {\n
constructor
=== js(\"Object\") -> PrimitiveClasses.anyClass\n
constructor === js(\"Error\") ->
PrimitiveClasses.throwableClass\n
else -> {\n
val jsClass: JsClass<T> =
constructor\n
getOrCreateKClass(jsClass)\n
}\n
}\n
}\n
}\n
}\n }.unsafeCast<KClass<T>>()\n\nprivate fun <T : Any> getOrCreateKClass(jClass: JsClass<T>):
KClass<T> {\n if (jClass === js(\"String\")) return PrimitiveClasses.stringClass.unsafeCast<KClass<T>>()\n\n
val metadata = jClass.asDynamic().`$metadata$`\n\n return if (metadata != null) {\n
if (metadata.`$kClass$`
== null) {\n
val kClass = SimpleKClassImpl(jClass)\n
metadata.`$kClass$` = kClass\n
kClass\n
} else {\n
metadata.`$kClass$`\n
}\n } else {\n
SimpleKClassImpl(jClass)\n }\n}\n","/*\n *
Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is

Open Source Used In IOx-FogDirector-AC18 1.17.0 2338


governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.js

/**
 */
@Suppress("NOT_DOCUMENTED")
public external class RegExp(pattern: String, flags: String? = definedExternally) {

    public fun test(str: String): Boolean

    public fun exec(str: String): RegExpMatch?

    public override fun toString(): String

    /**
     * The lastIndex is a read/write integer property of regular expressions that specifies the index at which to start the next match.
     * @return the new value of lastIndex
     */
    public var lastIndex: Int

    public val global: Boolean
    public val ignoreCase: Boolean
    public val multiline: Boolean

    public override fun toString(): String

    /**
     * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning of the input string.
     * @return the new value of lastIndex
     */
    public fun reset() {
        lastIndex = 0
    }

    // TODO: Inherit from array or introduce asArray() extension
    /**
     * Represents the return value of [RegExp.exec].
     */
    @Suppress("NOT_DOCUMENTED")
    public external interface RegExpMatch {
        public val index: Int
        public val input: String
        public val length: Int
    }

    /**
     * Returns the entire text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis at the given index.
     * @return the text matched by the capturing parenthesis
     */

    /**
     * Converts the result of [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent element contains the text matched by each capturing parenthesis.
     * @return the array of strings
     */
    public inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()

/**
 * Represents the return value of [RegExp.test] and [RegExp.exec] calls that match the entire string.
 */
@Suppress("NOT_DOCUMENTED")
public external interface RegExpMatchAll {
    public val regex: RegExp
    public val input: String
    public val matches: Array<out RegExpMatch

    /**
     * Returns the entire text matched by [RegExp.test] and [RegExp.exec] calls that match the entire string.
     * @return the entire text matched by [RegExp.test] and [RegExp.exec] calls that match the entire string
     */
    public fun toString(): String

    /**
     * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning of the input string.
     */
    public fun reset() {
        lastIndex = 0
    }

    // TODO: Inherit from array or introduce asArray() extension
    /**
     * Represents the return value of [RegExp.exec].
     */
    @Suppress("NOT_DOCUMENTED")
    public external interface RegExpMatch {
        public val index: Int
        public val input: String
        public val length: Int
    }

    /**
     * Returns the entire text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis at the given index.
     * @return the text matched by the capturing parenthesis
     */

    /**
     * Converts the result of [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent element contains the text matched by each capturing parenthesis.
     * @return the array of strings
     */
    public inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()

/**
 * Represents the return value of [RegExp.test] and [RegExp.exec] calls that match the entire string.
 */
@Suppress("NOT_DOCUMENTED")
public external interface RegExpMatchAll {
    public val regex: RegExp
    public val input: String
    public val matches: Array<out RegExpMatch

    /**
     * Returns the entire text matched by [RegExp.test] and [RegExp.exec] calls that match the entire string.
     * @return the entire text matched by [RegExp.test] and [RegExp.exec] calls that match the entire string
     */
    public fun toString(): String

    /**
     * Resets the regular expression so that subsequent [RegExp.test] and [RegExp.exec] calls will match starting with the beginning of the input string.
     */
    public fun reset() {
        lastIndex = 0
    }

    // TODO: Inherit from array or introduce asArray() extension
    /**
     * Represents the return value of [RegExp.exec].
     */
    @Suppress("NOT_DOCUMENTED")
    public external interface RegExpMatch {
        public val index: Int
        public val input: String
        public val length: Int
    }

    /**
     * Returns the entire text matched by [RegExp.exec] if the [index] parameter is 0, or the text matched by the capturing parenthesis at the given index.
     * @return the text matched by the capturing parenthesis
     */

    /**
     * Converts the result of [RegExp.exec] to an array where the first element contains the entire matched text and each subsequent element contains the text matched by each capturing parenthesis.
     * @return the array of strings
     */
    public inline fun RegExpMatch.asArray(): Array<out String?> = unsafeCast<Array<out String?>>()
*Note that the reverse operation may produce new surrogate pairs that were unpaired low-surrogates and high-surrogates before the operation.*

*For example, reversing `"\uDC00\uD800"` produces `"\uD800\uDC00"` which is a valid surrogate pair.*

*An actual fun reverse(): StringBuilder {
    var reversed = ""
    var index = string.length - 1
    while (index >= 0) {
        val low = string[index--]
        if (low.isLowSurrogate()) {
            index += 1
            reversed = reversed + low
        } else {
            reversed = reversed + high + low
        }
    }
    string = reversed
    return this
}

*Append the string representation of the specified object [value] to this string builder and returns this instance.*

*The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method, and then that string was appended to this string builder.*

*An actual fun append(value: Any?): StringBuilder {
    string += value.toString()
    return this
}

*Append the string representation of the specified boolean [value] to this string builder and returns this instance.*

*The capacity is the maximum length this string builder can have before an allocation occurs. In Kotlin/JS implementation of StringBuilder the value returned from this method may not indicate the actual size of the backing storage.*

*In Kotlin/JS implementation of StringBuilder the size of the backing storage is not extended to comply the given [minimumCapacity].

*Otherwise, this method takes no action and simply returns.*

*In Kotlin/JS implementation of StringBuilder the size of the backing storage is not extended to comply the given [minimumCapacity].

*This method has no effect on the further performance of operations.*

*Ensures that the capacity of this string builder is at least equal to the specified [minimumCapacity].

*If the current capacity is less than the [minimumCapacity], a new backing storage is allocated with greater capacity.

*Otherwise, this method takes no action and simply returns.*

*In Kotlin/JS implementation of StringBuilder the size of the backing storage is not extended to comply the given [minimumCapacity].

*Thus calling this method has no effect on the further performance of operations.*

*The last occurrence of the specified [string], starting at the specified [startIndex], toward the beginning.*

*The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method, and then that string was appended to this string builder.*

*An actual fun indexOf(value: String): Int = this.string.asDynamic().indexOf(string, startIndex)

*The overall effect is exactly as if the [value] were converted to a string by the `value.toString()` method, and then that string was appended to this string builder.*

*An actual fun lastIndexOf(string: String, startIndex: Int):

*Returns `-1` if the specified [string] does not occur in this string builder starting at the specified [startIndex].

*An actual fun lastIndexOfOf(string: String, startIndex: Int): Int = this.string.asDynamic().lastIndexOfOf(string, startIndex)
Returns a new [String] that contains characters in this string builder at [startIndex] (inclusive) and up to the [endIndex] (exclusive).

Throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] is out of range of this string builder indices or when `startIndex > endIndex`.

Attempted to reduce storage used for this string builder.

If the backing storage of this string builder is larger than necessary to hold its current contents, then it may be resized to become more space efficient.

Calling this method may, but is not required to, affect the value of the [capacity] property.

In Kotlin/JS implementation of StringBuilder the size of the backing storage is always equal to the length of the string builder.

If this.asDynamic().capacity !== undefined then this.asDynamic().capacity = length

Override fun toString(): String = string

public fun clear(): StringBuilder {
    string = ""
    return this
}

public operator fun set(index: Int, value: Char) {

    this.string = this.string.substring(0, index) + value + this.string.substring(index + 1)
}

public fun setRange(startIndex: Int, endIndex: Int, value: String): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)
    return this
}

private fun checkReplaceRange(startIndex: Int, endIndex: Int, length: Int) {
    if (startIndex < 0 || startIndex > length) {
        throw IndexOutOfBoundsException("startIndex: $startIndex, length: $length")
    }
    if (startIndex > endIndex) {
        throw IllegalArgumentException("startIndex($startIndex) > endIndex($endIndex)")
    }

    val startChar = this.string[startIndex]
    val endChar = this.string[endIndex]

    if (startIndex == endIndex)
        return

    val range = this.string.substring(startIndex, endIndex + 1)
    val newRange = range.substring(0, startIndex) + value + range.substring(startIndex + 1)

    this.string = newRange
}

public fun deleteAt(index: Int): StringBuilder {
    this.string = this.string.substring(0, index) + this.string.substring(index + 1)
    return this
}

public fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + this.string.substring(endIndex)
    return this
}

public fun setRange(startIndex, endIndex: Int, value: String): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)
    return this
}

public fun deleteRange(startIndex: Int, endIndex: Int): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + this.string.substring(endIndex)
    return this
}

private fun checkReplaceRange(startIndex, endIndex, length: Int) {
    if (startIndex < 0 || startIndex > length) {
        throw IndexOutOfBoundsException("startIndex: $startIndex, length: $length")
    }
    if (startIndex > endIndex) {
        throw IllegalArgumentException("startIndex($startIndex) > endIndex($endIndex)")
    }

    val startChar = this.string[startIndex]
    val endChar = this.string[endIndex]

    if (startIndex == endIndex)
        return

    val range = this.string.substring(startIndex, endIndex + 1)
    val newRange = range.substring(0, startIndex) + value + range.substring(startIndex + 1)

    this.string = newRange
}

public fun deleteAt(index: Int): StringBuilder {
    this.string = this.string.substring(0, index) + this.string.substring(index + 1)
    return this
}

public fun deleteRange(startIndex, endIndex: Int, value: String): StringBuilder {
    checkReplaceRange(startIndex, endIndex, length)

    this.string = this.string.substring(0, startIndex) + value + this.string.substring(endIndex)
    return this
}

private fun checkReplaceRange(startIndex, endIndex, length: Int) {
    if (startIndex < 0 || startIndex > length) {
        throw IndexOutOfBoundsException("startIndex: $startIndex, length: $length")
    }
    if (startIndex > endIndex) {
        throw IllegalArgumentException("startIndex($startIndex) > endIndex($endIndex)")
    }

    val startChar = this.string[startIndex]
    val endChar = this.string[endIndex]

    if (startIndex == endIndex)
        return

    val range = this.string.substring(startIndex, endIndex + 1)
    val newRange = range.substring(0, startIndex) + value + range.substring(startIndex + 1)

    this.string = newRange
}
* @param destinationOffset the position in the array to copy to, 0 by default.
* @param startIndex the beginning (inclusive) of the range to copy, 0 by default.
* @param endIndex the end (exclusive) of the range to copy, length of this string builder by default.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of this string builder indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].
* or when that index is out of the [destination] array indices range.
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public fun toCharArray(destination: CharArray, destinationOffset: Int = 0, startIndex: Int = 0, endIndex: Int = this.length) {
    AbstractList.checkBoundsIndexes(destinationOffset, destinationOffset + endIndex - startIndex, destination.size)
    var dstIndex = destinationOffset
    for (index in startIndex until endIndex) {
        destination[dstIndex++] = string[index]
    }
}

    /**
     * Appends characters in a subarray of the specified character array [value] to this string builder and returns this instance.
* @param value the array from which characters are appended.
* @param startIndex the beginning (inclusive) of the subarray to append.
* @param endIndex the end (exclusive) of the subarray to append.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when the subrange doesn't fit into the [destination] array starting at the specified [destinationOffset].
* or when that index is out of the [destination] array indices range.
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public fun appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder {
        string += value.concatToString(startIndex, endIndex)
        return this
    }

    /**
     * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.
* @param value the character sequence from which a subsequence is appended. If [value] is `null`, then characters will be appended as if [value] contained the four characters `"null"`.
* @param startIndex the beginning (inclusive) of the subsequence to append.
* @param endIndex the end (exclusive) of the subsequence to append.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException if [index] is less than zero or greater than the length of this string builder.
* @SinceKotlin("1.3")
* @ExperimentalStdlibApi
* public fun appendRange(value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder {
        val stringCsq = value.toString()
        AbstractList.checkBoundsIndexes(startIndex, endIndex, stringCsq.length)
        string += stringCsq.substring(startIndex, endIndex)
        return this
    }

    /**
     * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
* @param index the position in this string builder to insert at.
* @param value the array from which characters are inserted.
* @param startIndex the beginning (inclusive) of the subarray to insert.
* @param endIndex the end (exclusive) of the subarray to insert.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when characters will be inserted as if [value] contained the four characters `"null"`.
* @param index the position in this string builder to insert at.
* @param value the array from which characters are inserted.
* @param startIndex the beginning (inclusive) of the subarray to insert.
* @param endIndex the end (exclusive) of the subarray to insert.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when characters will be inserted as if [value] contained the four characters `"null"`.
* @param index the position in this string builder to insert at.
* @param value the array from which characters are inserted.
* @param startIndex the beginning (inclusive) of the subsequence to insert.
* @param endIndex the end (exclusive) of the subsequence to insert.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when characters will be inserted as if [value] contained the four characters `"null"`.
* @param index the position in this string builder to insert at.
* @param value the array from which characters are inserted.
* @param startIndex the beginning (inclusive) of the subsequence to insert.
* @param endIndex the end (exclusive) of the subsequence to insert.
* @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
* @throws IndexOutOfBoundsException when characters will be inserted as if [value] contained the four characters `"null"`.
*/
The content of this string builder making it empty and returns this instance.

*Throws IndexOutOfBoundsException if [index] is out of bounds of this string builder.*

*Throws IndexOutOfBoundsException or [IllegalArgumentException] if [startIndex] is less than zero, greater than the length of this string builder, or [startIndex] > [endIndex].*

*Strings.clearStringBuilder*
@param endIndex the end (exclusive) of the subarray to append.
*Throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.

@SinceKotlin("1.3")
@ExperimentalStdlibApi
@Suppress("EXTENSION_SHADOWED_BY_MEMBER",
  "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.appendRange(value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)

/**
 * Appends a subsequence of the specified character sequence [value] to this string builder and returns this instance.
 * @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@Suppress("EXTENSION_SHADOWED_BY_MEMBER",
  "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.appendRange(value: CharArray?, startIndex: Int, endIndex: Int): StringBuilder =
    this.appendRange(value, startIndex, endIndex)

/**
 * Inserts characters in a subarray of the specified character array [value] into this string builder at the specified [index] and returns this instance.
 * The inserted characters go in same order as in the [value] array, starting at [index].
 * @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] array indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@Suppress("EXTENSION_SHADOWED_BY_MEMBER",
  "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.insertRange(index: Int, value: CharArray, startIndex: Int, endIndex: Int): StringBuilder =
    this.insertRange(index, value, startIndex, endIndex)

/**
 * Inserts characters in a subsequence of the specified character sequence [value] into this string builder at the specified [index] and returns this instance.
 * The inserted characters go in same order as in the [value] character sequence, starting at [index].
 * @throws IndexOutOfBoundsException or IllegalArgumentException when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@Suppress("EXTENSION_SHADOWED_BY_MEMBER",
  "NOTHING_TO_INLINE")
public actual inline fun StringBuilder.insertRange(index: Int, value: CharSequence?, startIndex: Int, endIndex: Int): StringBuilder =
    this.insertRange(index, value, startIndex, endIndex)

package kotlin.text

// actually \s is enough to match all whitespace, but \xA0 added because of different regexp behavior of Rhino used in Selenium tests
public actual inline fun Char.isWhitespace(): Boolean =
toString().matches("[\s\xA0]")

@kotlin.internal.InlineOnly
public actual inline fun Char.toLowerCase(): Char =
js("String.fromCharCode")(toInt()).toLowerCase().charCodeAt(0).unsafeCast<Int>().toChar()

@kotlin.internal.InlineOnly
public actual inline fun Char.toUpperCase(): Char =
js("String.fromCharCode")(toInt()).toUpperCase().charCodeAt(0).unsafeCast<Int>().toChar()
public actual fun Char.isHighSurrogate(): Boolean = this in
Char.MIN_HIGH_SURROGATE..Char.MAX_HIGH_SURROGATE

public actual fun Char.isLowSurrogate(): Boolean = this in
Char.MIN_LOW_SURROGATE..Char.MAX_LOW_SURROGATE

public actual fun String.toBoolean(): Boolean = toLowerCase() == "true"

public actual fun String.toByte(): Byte = toByteOrNull() ?: numberFormatError(this)

public actual fun String.toByte(radix: Int): Byte = toByteOrNull(radix) ?:
numberFormatError(this)

public actual fun String.toShort(): Short = toShortOrNull() ?: numberFormatError(this)

public actual fun String.toShort(radix: Int): Short = toShortOrNull(radix) ?: numberFormatError(this)

public actual fun String.toInt(): Int = toIntOrNull() ?: numberFormatError(this)

public actual fun String.toInt(radix: Int): Int = toIntOrNull(radix) ?: numberFormatError(this)

public actual fun String.toLong(): Long = toLongOrNull() ?: numberFormatError(this)

public actual fun String.toLong(radix: Int): Long = toLongOrNull(radix) ?: numberFormatError(this)

public actual fun String.toDouble(): Double = (+(this.asDynamic())).unsafeCast<Double>().also {
    if (it.isNaN() && !this.isNaN() || it == 0.0 && this.isBlank())
        numberFormatError(this)
}

public actual inline fun String.toFloat(): Float =
toDouble().unsafeCast<Float>().unsafeCast<Double>().takeIf {
    | it >= 0.0 && !it.isNaN() && !this.isNaN() || it == 0.0 && !this.isBlank()|
} ?: numberFormatError(this)

public actual inline fun String.toDoubleOrNull(): Double? =
toDouble().unsafeCast<Double>().unsafeCast<String>().takeIf {
    | it >= 0.0 && !it.isNaN() && !this.isNaN() || it == 0.0 && !this.isBlank()|
} ?: numberFormatError(this)

public actual fun Byte.toString(radix: Int): String = this.toInt().toString(radix)

public actual fun Short.toString(radix: Int): String = this.toInt().toString(radix)

public actual fun Double.toString(): String = (+(this.asDynamic())).unsafeCast<Double>().takeIf {
    | it >= 0.0 && !it.isNaN() && !this.isNaN() || it == 0.0 && !this.isBlank()|
} ?: numberFormatError(this)

public actual fun String.toDouble(): Double =
toDouble().unsafeCast<String>().takeIf {
    | it >= 0.0 && !it.isNaN() && !this.isNaN() || it == 0.0 && !this.isBlank()|
} ?: numberFormatError(this)

Open Source Used In IOx-FogDirector-AC18 1.17.0 2346
inline fun Short.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this [Int] value in the specified [radix].
 */
@SinceKotlin("1.2")
public actual fun Int.toString(radix: Int): String =

/**
 * Checks whether the given [radix] is valid radix for string to number and number to string conversion.
 */

/**
 * Represents a compiled regular expression.
 */
public actual class Regex actual constructor(pattern: String, options: Set<RegexOption>) {

    /** Creates a regular expression from the specified [pattern] string and the default options. */
    public actual constructor(pattern: String) : this(pattern, emptySet())

    /** The pattern string of this regular expression. */
    public actual val pattern:

    /** The set of options that were used to create this regular expression. */
    public actual val options: Set<RegexOption> = options.toSet()

    private val nativePattern: RegExp = RegExp(pattern, options.map {
        it.value
    } + "g")

    /** Indicates whether the regular expression matches the entire [input]. */
    public actual infix fun matches(input: CharSequence): Boolean {
        nativePattern.reset()
        val match = nativePattern.exec(input.toString())
        return match != null && match.index == 0 &&
    }

    /** Indicates whether the regular expression can find at least one match in the specified [input]. */
    public actual fun containsMatchIn(input: CharSequence): Boolean {
        nativePattern.reset()
        return nativePattern.test(input.toString())
    }

    /** Returns a sequence of all occurrences of a regular expression within the [input] string, beginning at the specified [startIndex]. */
    public actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> =
        generateSequence({ find(input, startIndex) },
            { match -> match.next() })

    /** Attempts to match the entire [input] CharSequence against the pattern. */
    public actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? =
        nativePattern.exec(input.toString())

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun findNext(input: CharSequence, startIndex: Int = 0): MatchResult? =
        nativePattern.exec(input.toString())

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun generateSequence( { find(input, startIndex) },
            { match -> match.next() })

    return match != null && match.index == 0 &&
    nativePattern.lastIndex == input.length
}

/** Indicates whether the regular expression can find at least one match in the specified [input]. */
    public actual fun containsMatchIn(input: CharSequence): Boolean {
        nativePattern.reset()
        return nativePattern.test(input.toString())
    }

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? =
        nativePattern.findNext(input.toString(), startIndex)

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> =
        generateSequence( { find(input, startIndex) },
            { match -> match.next() })

    return match != null && match.index == 0 &&
    nativePattern.lastIndex == input.length
}

/** Indicates whether the regular expression can find at least one match in the specified [input]. */
    public actual fun containsMatchIn(input: CharSequence): Boolean {
        nativePattern.reset()
        return nativePattern.test(input.toString())
    }

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun find(input: CharSequence, startIndex: Int = 0): MatchResult? =
        nativePattern.findNext(input.toString(), startIndex)

@Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun findAll(input: CharSequence, startIndex: Int = 0): Sequence<MatchResult> =
        generateSequence( { find(input, startIndex) },
            { match -> match.next() })

    return match != null && match.index == 0 &&
    nativePattern.lastIndex == input.length
}
actual fun matchEntire(input: CharSequence): MatchResult? {
    if (pattern.startsWith('^') && pattern.endsWith('$'))
        return find(input)
    else
        return Regex("^${pattern.trimStart('^').trimEnd('$')}$", options).find(input)
}

    /**
     * Replaces all occurrences of this regular expression in the specified [input] string with specified [replacement] expression.
     */
    public actual fun replace(input: CharSequence, replacement: String): String = input.toString().nativeReplace(nativePattern, replacement)

    /**
     * Replaces all occurrences of this regular expression in the specified [input] string with the result of
     * the given function [transform] that takes [MatchResult] and returns a string to be used as a \n * replacement for that match.
     */
    public actual inline fun replace(input: CharSequence, transform: (MatchResult) -> CharSequence): String {
        var match = find(input)
        if (match == null) return input.toString()

        var lastStart = 0
        val length = input.length
        val sb = StringBuilder(length)
        do {
            val foundMatch = match!!
            sb.append(input, lastStart, foundMatch.range.start)
            sb.append(transform(foundMatch))
            lastStart = foundMatch.range.endInclusive + 1
            match = foundMatch.next()
        } while (lastStart < length && match != null)

        if (lastStart < length) {
            sb.append(input, lastStart, length)
        }

        return sb.toString()
    }

    /**
     * Replaces the first occurrence of this regular expression in the specified [input] string with specified [replacement] expression.
     */
    public actual fun replaceFirst(input: CharSequence, replacement: String): String {
        val nonGlobalOptions = options.map { it.value }.joinToString(separator = "")
        return input.toString().nativeReplace(RegExp(pattern, nonGlobalOptions), replacement)
    }

    /**
     * Splits the [input] CharSequence around matches of this regular expression.
     *
     * @param limit Non-negative value specifying the maximum number of substrings
     * the string can be split to. \n * Zero by default means no limit is set.
     */
    @Suppress("ACTUAL_FUNCTION_WITH_DEFAULT_ARGUMENTS")
    public actual fun split(input: CharSequence, limit: Int = 0): List<String> {
        require(limit >= 0) { "Limit must be non-negative, but was $limit" }
        val matches = findAll(input).let { if (limit == 0) it else it.take(limit - 1) }
        val result = mutableListOf<String>()
        var lastStart = 0
        for (match in matches) {
            result.add(input.subSequence(lastStart, match.range.start).toString())
            lastStart = match.range.endInclusive + 1
        }
        result.add(input.subSequence(lastStart, input.length).toString())
        return result
    }

    */
    /**
     * Returns the string representation of this regular expression, namely the [pattern] of this regular
     * expression.
     */
    public override fun toString(): String = nativePattern.toString()

    actual companion object {
        /**
         * Returns a regular expression that
         * matches the specified [literal] string literally.
         * No characters of that string will have special meaning when searching for an occurrence of the regular expression.
         */
        public actual fun fromLiteral(literal: String): Regex = Regex(escape(literal))

        /**
         * Returns a regular expression pattern string that matches the
         * specified [literal] string literally.
         * No characters of that string will have special meaning when searching for an occurrence of the regular expression.
         */
        public actual fun escape(literal: String): String = literal.nativeReplace(patternEscape, "\\$&")

        /**
         * Returns a literal replacement expression for the
         * specified [literal] string.
         * No characters of that string will have special meaning when it is used as a
         * replacement string in [Regex.replace] function.
         */
        public actual fun escapeReplacement(literal: String): String = literal.nativeReplace(replacementEscape, "****")
    }
}
DeprecationLevel.HIDDEN)\n@JsName("Regex_sb3q2\$\")\npublic fun Regex_0(pattern: String, option: RegexOption): Regex = Regex(pattern, setOf(option))\n/**\n * Creates a regular expression from the specified [pattern] string and the default options.  *
 */\n@JsName("Regex_61zpoe\$\")\npublic fun Regex_1(pattern: String): Regex = Regex(pattern, emptySet())


/** Creates a regular expression from the specified [pattern] string and the default options.  */\n@Deprecated("Provided for binary compatibility", level = DeprecationLevel.HIDDEN)\n@JsName("Regex_61zpoe\$\")\npublic fun Regex_1(pattern: String): Regex = Regex(pattern, emptySet())
actual fun String.decapitalize(): String { if (!isEmpty()) substring(0, 1).toLowerCase() + substring(1) else this

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.text

fun replace(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) "gi" else "g"),
Regex.escapeReplacement(newValue))

fun replace(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "i" else ""),
newChar.toString())

fun replaceFirst(oldValue: String, newValue: String, ignoreCase: Boolean = false): String =
nativeReplace(RegExp(Regex.escape(oldValue), if (ignoreCase) "i" else ""),
Regex.escapeReplacement(newValue))

fun replaceFirst(oldChar: Char, newChar: Char, ignoreCase: Boolean = false): String =
nativeReplace(RegExp(Regex.escape(oldChar.toString()), if (ignoreCase) "i" else ""),
newChar.toString())

@UseExperimental(ExperimentalStdlibApi::class) private fun unpackage kotlin.text

}"
0xE0 != 0xA0) {
    // Non-shortest form
    return malformed(0, index, throwOnMalformed)
} else if (byte1 and 0xF == 0x0D) {
    if (byte2 and 0xE0 != 0x80) {
        // Surrogate code point
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    } else if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    } else if (byte2 and 0xE0 != 0x80) {
        // Surrogate code point
        return malformed(0, index, throwOnMalformed)
    }
}

if (byte2 and 0xC0 != 0x80) {
    return malformed(0, index, throwOnMalformed)
}

if (index + 1 == endIndex) {
    return malformed(1, index, throwOnMalformed)
}
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    return (byte1 shl 12) xor (byte2 shl 6) xor byte3 xor -0x1E080
}

/**
 * Returns code point corresponding to UTF-8 sequence of four bytes, * where the first byte
 * of the sequence is the [byte1] and the others are in the [bytes] array starting from the [index].
 * Returns a non-positive value indicating number of bytes from [bytes] included in malformed sequence
 * if the sequence is malformed and [throwOnMalformed] is false.
 *
 * @param bytes the bytes array.
 * @param byte1 the first byte of sequence
 * @param index the starting index in [bytes]
 * @param endIndex index of the end of sequence.
 * @param throwOnMalformed whether to throw on malformed cha sequence or return.
 * @throws CharacterCodingException if the sequence of four bytes is malformed and [throwOnMalformed] is true.
 * @private fun codePointFrom4(bytes: ByteArray, byte1: Int, index: Int, endIndex: Int, throwOnMalformed: Boolean): Int
 */
    if (index >= endIndex) {
        malformed(0, index, throwOnMalformed)
    }

    val byte2 = bytes[index].toInt()
    if (byte1 and 0xF == 0x0) {
        if (byte2 and 0xF0 <= 0x80) {
            // Non-shortest form
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF == 0x4) {
        if (byte2 and 0xF0 != 0x80) {
            // Out of Unicode code points domain (larger than U+10FFFF)
            return malformed(0, index, throwOnMalformed)
        }
    } else if (byte1 and 0xF > 0x4) {
        return malformed(0, index, throwOnMalformed)
    } else if (byte2 and 0xC0 != 0x80) {
        return malformed(0, index, throwOnMalformed)
    }

    if (index + 1 == endIndex) {
        return malformed(1, index, throwOnMalformed)
    }
    val byte3 = bytes[index + 1].toInt()
    if (byte3 and 0xC0 != 0x80) {
        return malformed(1, index, throwOnMalformed)
    }

    if (index + 2 == endIndex) {
        return malformed(2, index, throwOnMalformed)
    }
    val byte4 = bytes[index + 2].toInt()
    if (byte4 and 0xC0 != 0x80) {
        return malformed(2, index, throwOnMalformed)
    }
    return (byte1 shl 18) xor (byte2 shl 12) xor (byte3 shl 6) xor byte4 xor 0x381F80
}

/**
 * Maximum number of bytes needed to encode a single char.
 *
 * Code points in `0..0x7F` are encoded in a single byte.
 * Code points in `0x80..0x7FF` are encoded in two bytes.
 * Code points in `0x800..0xD7FF` or in `0xE000..0xFFFF` are encoded in three bytes.
 */
private const val MAX_BYTES_PER_CHAR = 3

/**
 * The byte sequence a malformed UTF-16 char sequence is replaced by.
 */
private val REPLACEMENT_BYTE_SEQUENCE: ByteArray = byteArrayOf(0xEF.toByte(),
    0xBF.toByte(), 0xBD.toByte())

/**
 * Encodes the [string] using UTF-8 and returns the resulting [ByteArray].
 *
 * @param string the string to encode.
 * @param startIndex the start offset (inclusive) of the substring to encode.
 * @param endIndex the end offset (exclusive) of the substring to encode.
 * @param throwOnMalformed whether to throw on malformed char sequence or replace by the
 * [REPLACEMENT_BYTE_SEQUENCE].
 * @throws CharacterCodingException if the char sequence is malformed and [throwOnMalformed] is true.
 */
internal fun encodeUtf8(string: String, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): ByteArray {
    require(startIndex >= 0 && endIndex <= string.length && startIndex <= endIndex)

    val bytes = ByteArray((endIndex - startIndex) * MAX_BYTES_PER_CHAR)
    var byteIndex = 0
    var charIndex = startIndex

    while (charIndex < endIndex) {
        val code = string[charIndex++].toInt()
        when {
            code < 0x80 ->
                bytes[byteIndex++] = code.toByte()
            code < 0x800 -> {
                bytes[byteIndex++] = ((code shr 6) or 0xC0).toByte()
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
            }
            code < 0xD800 ->
                bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                bytes[byteIndex++] = ((code shr 12) or 0xE0).toByte()
                bytes[byteIndex++] = ((code shr 6) or 0x80).toByte()
            else -> {
                val codePoint = codePointFromSurrogate(string, charIndex, throwOnMalformed)
                if (codePoint <= 0) {
                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[0]
                    bytes[byteIndex++] = REPLACEMENT_BYTE_SEQUENCE[1]
                } else {
                    bytes[byteIndex++] = ((code shr 12) or 0x80).toByte()
                    bytes[byteIndex++] = ((code shr 6) or 0x80).toByte()
                    bytes[byteIndex++] = ((code and 0x3F) or 0x80).toByte()
                }
            }
        }
    }
    return bytes
}
REPLACEMENT_BYTE_SEQUENCE[2]

bytes[byteIndex++] = ((codePoint shr 18) or 0xF0).toByte()
bytes[byteIndex++] = (((codePoint shr 12) and 0x3F) or 0x80).toByte()
bytes[byteIndex++] = (((codePoint shr 6) and 0x3F) or 0x80).toByte()
bytes[byteIndex++] = (codePoint and 0x3F) or 0x80).toByte()
charIndex++

return if (bytes.size == byteIndex) bytes else bytes.copyOf(byteIndex)
}

/**
 * The character a malformed UTF-8 byte sequence is replaced by.
 */
private const val REPLACEMENT_CHAR = '\uFFFD'

/**
 * Decodes the UTF-8 [bytes] array and returns the resulting [String].
 */
internal fun decodeUtf8(bytes: ByteArray, startIndex: Int, endIndex: Int, throwOnMalformed: Boolean): String {
    require(startIndex >= 0 && endIndex <= bytes.size && byteIndex <= endIndex)

    var byteIndex = startIndex
    while (byteIndex < endIndex) {
        val byte = bytes[byteIndex++].toInt()
        when {
            byte >= 0 ->
                stringBuilder.append(byte.toChar())
            byte shr 5 == -2 -> {
                val code = codePointFrom2(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                if (code <= 0) {
                    stringBuilder.append(REPLACEMENT_CHAR)
                    byteIndex += -code
                } else {
                    stringBuilder.append(code.toChar())
                    byteIndex += 1
                }
            }
            byte shr 4 == -2 - 1 -> {
                val code = codePointFrom3(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                if (code <= 0) {
                    stringBuilder.append(REPLACEMENT_CHAR)
                    byteIndex += -code
                } else {
                    stringBuilder.append(code.toChar())
                    byteIndex += 2
                }
            }
            byte shr 3 == -2 -> {
                val code = codePointFrom4(bytes, byte, byteIndex, endIndex, throwOnMalformed)
                if (code <= 0) {
                    stringBuilder.append(REPLACEMENT_CHAR)
                    byteIndex += -code
                } else {
                    val high = (code - 0x10000) shr 10 or 0xD800
                    val low = (code and 0x3FF) or 0xDC00
                    stringBuilder.append(high.toChar())
                    stringBuilder.append(low.toChar())
                    byteIndex += 3
                }
            }
            else -> {
                malformed(0, byteIndex, throwOnMalformed)
                stringBuilder.append(REPLACEMENT_CHAR)
                byteIndex += 4
            }
        } 
        byteIndex = byteIndex - byteshr
    }
    return stringBuilder.toString()
}
override fun markNow(): TimeMark = actualSource.markNow()
}

internal
external interface Process {
    fun hrtime(time: Array<Double> = definedExternally): Array<Double>
}

@SinceKotlin("1.3")
@ExperimentalTime
internal class HrTimeSource(val process: Process) : TimeSource {

    override fun markNow(): TimeMark = object : TimeMark() {
        val startedAt = process.hrtime()
        override fun elapsedNow(): Duration = process.hrtime(startedAt).let { (seconds, nanos) ->
            seconds.seconds + nanos.nanoseconds
        }
    }

    override fun toString(): String = "TimeSource(process.hrtime())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal class PerformanceTimeSource(val performance: Performance) : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = performance.now()
    override fun toString(): String = "TimeSource(self.performance.now())"
}

@SinceKotlin("1.3")
@ExperimentalTime
internal object DateNowTimeSource : AbstractDoubleTimeSource(unit = DurationUnit.MILLISECONDS) {
    override fun read(): Double = kotlin.js.Date.now()
    override fun toString(): String = "TimeSource(Date.now())"
}
`true` /* = false */
        get() = definedExternally
        set(value) =
        
    @kotlin.internal.InlineOnly
        val o = js("({})")
        o[`alpha`] = alpha
        o[`depth`] = depth
        o[`stencil`] = stencil
        o[`antialias`] = antialias
        o[`premultipliedAlpha`] = premultipliedAlpha
        o[`preserveDrawingBuffer`] = preserveDrawingBuffer
        o[`preferLowPowerToHighPerformance`] = preferLowPowerToHighPerformance
        o[`failIfMajorPerformanceCaveat`] = failIfMajorPerformanceCaveat
        return o
    }

    public external abstract class WebGLObject

    /**
     * Exposes the JavaScript
     * WebGLBuffer([https://developer.mozilla.org/en/docs/Web/API/WebGLBuffer] to Kotlin
     */
    public external abstract class WebGLBuffer : WebGLObject

    /**
     * Exposes the JavaScript
     * WebGLFramebuffer([https://developer.mozilla.org/en/docs/Web/API/WebGLFramebuffer] to Kotlin
     */
    public external abstract class WebGLFramebuffer : WebGLObject

    /**
     * Exposes the JavaScript
     * WebGLProgram([https://developer.mozilla.org/en/docs/Web/API/WebGLProgram] to Kotlin
     */
    public external abstract class WebGLProgram : WebGLObject

    /**
     * Exposes the JavaScript
     * WebGLRenderbuffer([https://developer.mozilla.org/en/docs/Web/API/WebGLRenderbuffer] to Kotlin
     */
    public
    external abstract class WebGLRenderbuffer : WebGLObject

    /**
     * Exposes the JavaScript
     */
    public external abstract class WebGLShader : WebGLObject

    /**
     * Exposes the JavaScript
     */
    public external abstract class WebGLTexture : WebGLObject

    @Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
    public external interface WebGLRenderingContextBase {
        val canvas: HTMLCanvasElement
        val drawingBufferWidth: Int
        val drawingBufferHeight: Int
        fun getContextAttributes(): WebGLContextAttributes?
        fun isContextLost(): Boolean
        fun getSupportedExtensions(): Array<String>
        fun getExtension(name: String): dynamic
        fun activeTexture(texture: Int)
        fun attachShader(program: WebGLProgram?, shader: WebGLShader?)
        fun bindAttribLocation(program: WebGLProgram?, index: Int, name: String)
        fun bindBuffer(target: Int, buffer: WebGLBuffer?)
        fun bindFramebuffer(target: Int, framebuffer: WebGLFramebuffer?)
        fun bindRenderbuffer(target: Int, renderbuffer: WebGLRenderbuffer?)
        fun bindTexture(target: Int, texture: WebGLTexture?)
        fun blendColor(red: Float, green: Float, blue: Float, alpha: Float)
        fun blendEquation(mode: Int)
        fun blendEquationSeparate(modeRGB: Int, modeAlpha: Int)
        fun blendFunc(sfactor: Int, dfactor: Int)
        fun blendFuncSeparate(srcRGB: Int, dstRGB: Int, srcAlpha: Int, dstAlpha: Int)
        fun bindData(target: Int, size: Int, usage: Int)
        fun bufferData(target: Int, data: BufferDataSource?, usage: Int)
        fun checkFramebufferStatus(target: Int)
        fun clearColor(red: Float, green: Float, blue: Float, alpha: Float)
        fun clearDepth(depth: Float)
        fun clearStencil(stencil: Int)
        fun colorMask(red: Boolean, green: Boolean, blue: Boolean, alpha: Boolean)
        fun compileShader(shader: WebGLShader?)
height: Int, format: Int, data: ArrayBufferView?)


fun createBuffer(): WebGLBuffer?

fun createFramebuffer(): WebGLFramebuffer?

fun createShader(type: Int): WebGLShader?

fun createTexture(): WebGLTexture?

fun cullFace(mode: Int)

fun deleteBuffer(buffer: WebGLBuffer?)

fun deleteFramebuffer(framebuffer: WebGLFramebuffer?)

fun deleteProgram(program: WebGLProgram?)

fun deleteRenderbuffer(renderbuffer: WebGLRenderbuffer?)

fun deleteShader(shader: WebGLShader?)

fun deleteTexture(texture: WebGLTexture?)

fun depthFunc(func: Int)

fun depthMask(flag: Boolean)

fun depthRange(zNear: Float, zFar: Float)

fun detachShader(program: WebGLProgram?, shader: WebGLShader?)

fun disable(cap: Int)

fun disableVertexAttribArray(index: Int)

fun drawArrays(mode: Int, first: Int, count: Int)

fun drawElements(mode: Int, count: Int, type: Int, offset: Int)

fun enable(cap: Int)

fun enableVertexAttribArray(index: Int)

fun finish()

fun flush()

fun framebufferRenderbuffer(target: Int, attachment: Int, renderbuffertarget: Int, renderbuffer: WebGLRenderbuffer?)


fun generateMipmap(target: Int)

fun getActiveAttrib(program: WebGLProgram?, index: Int): WebGLActiveInfo?

fun getActiveUniform(program: WebGLProgram?, index: Int): WebGLActiveInfo?

fun getAttachedShaders(program: WebGLProgram?): Array<WebGLShader>?

fun getAttribLocation(program: WebGLProgram?, name: String): Int

fun getBufferParameter(target: Int, pname: Int): Any?

fun getProgramParameter(target: Int, pname: Int): Any?

fun getProgramInfoLog(program: WebGLProgram?): String?

fun getProgramUniform(program: WebGLProgram?, location: Int): WebGLUniformLocation?

fun getTexParameter(target: Int, pname: Int): Any?

fun getUniform(program: WebGLProgram?, location: WebGLUniformLocation?): Any?

fun getVertexAttrib(index: Int, pname: Int): Any?

fun getVertexAttribOffset(index: Int, pname: Int): Int

fun hint(target: Int, mode: Int)

fun isBuffer(buffer: WebGLBuffer?): Boolean

fun isEnabled(cap: Int): Boolean

fun isFramebuffer(framebuffer: WebGLFramebuffer?): Boolean

fun isProgram(program: WebGLProgram?): Boolean

fun isRenderbuffer(renderbuffer: WebGLRenderbuffer?): Boolean

fun isShader(shader: WebGLShader?): Boolean

fun isTexture(texture: WebGLTexture?): Boolean

fun pixelStorei(pname: Int, param: Int)

fun polygonOffset(factor: Float, units: Float)

fun readPixels(x: Int, y: Int, width: Int, height: Int, format: Int, type: Int, pixels: ArrayBufferView?)

fun renderbufferStorage(target: Int, internalformat: Int, width: Int, height: Int)

fun sampleCoverage(value: Float, invert: Boolean)

fun stencilFunc(func: Int, ref: Int, mask: Int)

fun stencilFuncSeparate(face: Int, func: Int, ref: Int, mask: Int)

fun stencilMask(mask: Int)

fun stencilMaskSeparate(face: Int, mask: Int)

fun stencilOp(fail: Int, zfail: Int, zpass: Int)


fun texImage2D(target: Int, level: Int, internalformat: Int, format: Int, source: TexImageSource?)


fun uniform1f(location: WebGLUniformLocation?, value: Float)

fun uniform1fv(location: WebGLUniformLocation?, array: Float32Array)

fun uniform1fv(location: WebGLUniformLocation?, x: Float)

fun uniform1fv(location: WebGLUniformLocation?, y: Float)

fun uniform1fv(location: WebGLUniformLocation?, v: Float32Array)

fun uniform1fv(location: WebGLUniformLocation?, w: Float32Array)
TEXTURE18: Int
        val TEXTURE19: Int
        val TEXTURE20: Int
        val TEXTURE21: Int
        val
TEXTURE22: Int
        val TEXTURE23: Int
        val TEXTURE24: Int
        val TEXTURE25: Int
        val
TEXTURE26: Int
        val TEXTURE27: Int
        val TEXTURE28: Int
        val TEXTURE29: Int
        val
TEXTURE30: Int
        val TEXTURE31: Int
        val ACTIVE_TEXTURE: Int
        val REPEAT: Int

val CLAMP_TO_EDGE: Int
        val MIRRORED_REPEAT: Int
        val FLOAT_VEC2: Int
        val
FLOAT_VEC3: Int
        val FLOAT_VEC4: Int
        val INT_VEC2: Int
        val
INT_VEC3: Int
        val INT_VEC4: Int
        val BOOL: Int
        val
BOOL_VEC2: Int
        val BOOL_VEC3: Int
        val BOOL_VEC4: Int
        val
FLOAT_MAT2: Int
        val FLOAT_MAT3: Int
        val FLOAT_MAT4: Int
        val SAMPLER_2D: Int
        val SAMPLER_CUBE: Int
        val VERTEX_ATTRIB_ARRAY_ENABLED:
        val VERTEX_ATTRIB_ARRAY_SIZE: Int
        val VERTEX_ATTRIB_ARRAY_STRIDE:
        val VERTEX_ATTRIB_ARRAY_TYPE:
        val VERTEX_ATTRIB_ARRAY_NORMALIZED:
        val VERTEX_ATTRIB_ARRAY_POINTER:
        val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING:
        val IMPLEMENTATION_COLOR_READ_TYPE:
        val IMPLEMENTATION_COLOR_READ_FORMAT:
        val LOW_FLOAT: Int
        val MEDIUM_FLOAT: Int
        val HIGH_FLOAT: Int
        val LOW_INT: Int
        val MEDIUM_INT: Int
        val HIGH_INT: Int
        val FRAMEBUFFER: Int
        val RENDERBUFFER:
        val RGBA4: Int
        val RGB5_A1: Int
        val RGB565: Int
        val DEPTH_COMPONENT16:
        val STENCIL_INDEX:
        val STENCIL_INDEX8:
        val DEPTH_STENCIL:
        val RENDERBUFFER_WIDTH:
        val RENDERBUFFER_HEIGHT:
        val RENDERBUFFER_INTERNAL_FORMAT:
        val RENDERBUFFER_RED_SIZE:
        val RENDERBUFFER_GREEN_SIZE:
        val RENDERBUFFER_BLUE_SIZE:
        val RENDERBUFFER_DEPTH_SIZE:
        val RENDERBUFFER_STENCIL_SIZE:
        val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE:
        val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME:
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL:
        val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE:
        val COLOR_ATTACHMENT0:
        val DEPTH_ATTACHMENT:
        val STENCIL_ATTACHMENT:
        val DEPTH_STENCIL_ATTACHMENT:
        val NONE:
        val FRAMEBUFFER_COMPLETE:
        val FRAMEBUFFER_INCOMPLETE_ATTACHMENT:
        val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT:
        val FRAMEBUFFER_INCOMPLETE_DIMENSIONS:
        val FRAMEBUFFER_UNSUPPORTED:
        val FRAMEBUFFER_BINDING:
        val RENDERBUFFER_BINDING:
        val MAX_RENDERBUFFER_SIZE:
        val INVALID_FRAMEBUFFER_OPERATION:
        val UNPACK_FLIP_Y_WEBGL:
        val UNPACK_PREMULTIPLY_ALPHA_WEBGL:
        val CONTEXT_LOST_WEBGL:
        val BROWSER_DEFAULT_WEBGL:

/**
 * Exposes the JavaScript
[WebGLRenderingContext](https://developer.mozilla.org/en/docs/Web/API/WebGLRenderingContext) to Kotlin
 */
public external abstract class WebGLRenderingContext : WebGLRenderingContextBase, RenderingContext {

companion object {
        val DEPTH_BUFFER_BIT:
        val STENCIL_BUFFER_BIT:
        val COLOR_BUFFER_BIT:
        val POINTS:
        val LINE_LOOP:
        val LINE_STRIP:
        val TRIANGLES:
        val ZERO:
        val ONE:
        val SRC_COLOR:
        val ONE_MINUS_SRC_COLOR:
        val SRC_ALPHA:
        val ONE_MINUS_SRC_ALPHA:
        val DST_ALPHA:
        val ONE_MINUS_DST_ALPHA:
        val DST_COLOR:
        val ONE_MINUS_DST_COLOR:
        val SRC_ALPHA_SATURATE:
        val FUNC_ADD:
        val BLEND_EQUATION:
        val BLEND_EQUATION_RGB:
        val BLEND_EQUATION_ALPHA:
        val FUNC_REVERSE_SUBTRACT:
        val BLEND_SRC_RGB:
        val BLEND_DST_RGB:
        val BLEND_SRC_ALPHA:
        val BLEND_DST_ALPHA:

Open Source Used In IOx-FogDirector-AC18 1.17.0 2358
CONSTANT_COLOR: Int
ONE_MINUS_CONSTANT_COLOR: Int
CONSTANT_ALPHA: Int
ONE_MINUS_CONSTANT_ALPHA: Int
BLEND_COLOR: Int
ARRAY_BUFFER: Int
ELEMENT_ARRAY_BUFFER: Int
ARRAY_BUFFER_BINDING: Int
STREAM_DRAW: Int
DYNAMIC_DRAW: Int
BUFFER_SIZE: Int
BUFFER_USAGE: Int
CURRENT_VERTEX_ATTRIB: Int
FRONT: Int
BACK: Int
FRONT_AND_BACK: Int
CULL_FACE: Int
BLEND: Int
DITHER: Int
STENCIL_TEST: Int
DEPTH_TEST: Int
SCISSOR_TEST: Int
POLYGON_OFFSET_FILL: Int
SAMPLE_ALPHA_TO_COVERAGE: Int
SAMPLE_COVERAGE: Int
NO_ERROR: Int
INVALID_ENUM: Int
INVALID_VALUE: Int
INVALID_OPERATION: Int
OUT_OF_MEMORY: Int
CW: Int
CCW: Int
LINE_WIDTH: Int
ALIASED_POINT_SIZE_RANGE: Int
ALIASED_LINE_WIDTH_RANGE: Int
DEPTH_RANGE: Int
DEPTH_WRITEMASK: Int
DEPTH_CLEAR_VALUE: Int
DEPTH_FUNC: Int
STENCIL_CLEAR_VALUE: Int
STENCIL_FUNC: Int
STENCIL_FAIL: Int
STENCIL_PASS_DEPTH_FAIL: Int
STENCIL_PASS_DEPTH_PASS: Int
STENCIL_REF: Int
STENCIL_VALUE_MASK: Int
STENCIL_BACK_FUNC: Int
STENCIL_BACK_FAIL: Int
STENCIL_BACK_PASS_DEPTH_FAIL: Int
STENCIL_BACK_PASS_DEPTH_PASS: Int
STENCIL_BACK_REF: Int
STENCIL_BACK_VALUE_MASK: Int
STENCIL_BACK_WRITEMASK: Int
VIEWPORT: Int
SCISSOR_BOX: Int
COLOR_CLEAR_VALUE: Int
COLOR_WRITEMASK: Int
UNPACK_ALIGNMENT: Int
PACK_ALIGNMENT: Int
MAX_TEXTURE_SIZE: Int
MAX_VIEWPORT_DIMS: Int
SUBPIXEL_BITS: Int
RED_BITS: Int
GREEN_BITS: Int
BLUE_BITS: Int
ALPHA_BITS: Int
DEPTH_BITS: Int
STENCIL_BITS: Int
POLYGON_OFFSET_UNITS: Int
POLYGON_OFFSET_FACTOR: Int
TEXTURE_BINDING_2D: Int
SAMPLE_BUFFERS: Int
SAMPLES: Int
SAMPLE_COVERAGE_VALUE: Int
SAMPLE_COVERAGE_INVERT: Int
COMPRESSED_TEXTURE_FORMATS: Int
DONT_CARE: Int
FASTEST: Int
NICEST: Int
GENERATE_MIPMAP_HINT: Int
BYTE: Int
UNSIGNED_BYTE: Int
SHORT: Int
UNSIGNED_SHORT: Int
INT: Int
UNSIGNED_INT: Int
FLOAT: Int
DEPTH_COMPONENT: Int
ALPHA: Int
RGB: Int
RGBA: Int
LUMINANCE: Int
LUMINANCE_ALPHA: Int
UNSIGNED_SHORT_4_4_4_4: Int
UNSIGNED_SHORT_5_6_5: Int
UNSIGNED_SHORT_5_5_5_1: Int
FRAGMENT_SHADER: Int
VERTEX_SHADER: Int
MAX_VERTEX_ATTRIBS: Int
MAX_VERTEX_UNIFORM_VECTORS: Int
MAX_VARYING_VECTORS: Int
MAX_TEXTURE_IMAGE_UNITS: Int
MAX_FRAGMENT_UNIFORM_VECTORS: Int
SHADER_TYPE: Int
DELETE_STATUS: Int
VALIDATE_STATUS: Int
ATTACHED_SHADERS: Int
ACTIVE_UNIFORMS: Int
ACTIVE_ATTRIBUTES: Int
SHADING_LANGUAGE_VERSION: Int
CURRENT_PROGRAM: Int
NEVER: Int
LESS: Int
EQUAL: Int
LEQUAL: Int
GREATER: Int
NOTEQUAL: Int
GEQUAL: Int
ALWAYS: Int
KEEP: Int
REPLACE: Int
INCR: Int
DEC: Int
INVERT: Int
INCR_WRAP: Int
DEC_WRAP: Int
VERSION: Int
NEAREST: Int
LINEAR: Int
NEAREST_MIPMAP_NEAREST: Int
LINEAR_MIPMAP_NEAREST: Int
NEAREST_MIPMAP_LINEAR: Int
LINEAR_MIPMAP_LINEAR: Int
val TEXTURE_MAG_FILTER: Int
val TEXTURE_MIN_FILTER: Int
val TEXTURE_WRAP_S: Int
val TEXTURE_WRAP_T: Int
val TEXTURE_2D: Int
val TEXTURE: Int
val TEXTURE_CUBE_MAP: Int
val TEXTURE_BINDING_CUBE_MAP: Int
val TEXTURE_CUBE_MAP_POSITIVE_X: Int
val TEXTURE_CUBE_MAP_NEGATIVE_X: Int
val TEXTURE_CUBE_MAP_POSITIVE_Y: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Y: Int
val TEXTURE_CUBE_MAP_POSITIVE_Z: Int
val TEXTURE_CUBE_MAP_NEGATIVE_Z: Int
val MAX_CUBE_MAP_TEXTURE_SIZE: Int
val TEXTURE0: Int
val TEXTURE1: Int
val TEXTURE2: Int
val TEXTURE3: Int
val TEXTURE4: Int
val TEXTURE5: Int
val TEXTURE6: Int
val TEXTURE7: Int
val TEXTURE8: Int
val TEXTURE9: Int
val TEXTURE10: Int
val TEXTURE11: Int
val TEXTURE12: Int
val TEXTURE13: Int
val TEXTURE14: Int
val TEXTURE15: Int
val TEXTURE16: Int
val TEXTURE17: Int
val TEXTURE18: Int
val TEXTURE19: Int
val TEXTURE20: Int
val TEXTURE21: Int
val TEXTURE22: Int
val TEXTURE23: Int
val TEXTURE24: Int
val TEXTURE25: Int
val TEXTURE26: Int
val TEXTURE27: Int
val TEXTURE28: Int
val TEXTURE29: Int
val TEXTURE30: Int
val TEXTURE31: Int
val ACTIVE_TEXTURE: Int
val CLAMP_TO_EDGE: Int
val MIRRORED_REPEAT: Int
val FLOAT_VEC2: Int
val FLOAT_VEC3: Int
val FLOAT_VEC4: Int
val INT_VEC2: Int
val INT_VEC3: Int
val INT_VEC4: Int
val BOOL: Int
val BOOL_VEC2: Int
val BOOL_VEC3: Int
val BOOL_VEC4: Int
val SAMPLER_2D: Int
val SAMPLER_CUBE: Int
val VERTEX_ATTRIB_ARRAY_ENABLED: Int
val VERTEX_ATTRIB_ARRAY_SIZE: Int
val VERTEX_ATTRIB_ARRAY_STRIDE: Int
val VERTEX_ATTRIB_ARRAY_TYPE: Int
val VERTEX_ATTRIB_ARRAY_NORMALIZED: Int
val VERTEX_ATTRIB_ARRAY_POINTER: Int
val VERTEX_ATTRIB_ARRAY_BUFFER_BINDING: Int
val IMPLEMENTATION_COLOR_READ_TYPE: Int
val IMPLEMENTATION_COLOR_READ_FORMAT: Int
val LOW_FLOAT: Int
val MEDIUM_FLOAT: Int
val HIGH_FLOAT: Int
val LOW_INT: Int
val MEDIUM_INT: Int
val HIGH_INT: Int
val FRAMEBUFFER: Int
val RENDERBUFFER: Int
val RGBA4: Int
val RGB5_A1: Int
val RGB565: Int
val DEPTH_COMPONENT16: Int
val STENCIL_INDEX: Int
val STENCIL_INDEX8: Int
val DEPTH_STENCIL: Int
val RENDERBUFFER_WIDTH: Int
val RENDERBUFFER_HEIGHT: Int
val RENDERBUFFER_INTERNAL_FORMAT: Int
val RENDERBUFFER_RED_SIZE: Int
val RENDERBUFFER_GREEN_SIZE: Int
val RENDERBUFFER_BLUE_SIZE: Int
val RENDERBUFFER_ALPHA_SIZE: Int
val RENDERBUFFER_DEPTH_SIZE: Int
val RENDERBUFFER_STENCIL_SIZE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE: Int
val FRAMEBUFFER_ATTACHMENT_OBJECT_NAME: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL: Int
val FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE: Int
val COLOR_ATTACHMENT0: Int
val DEPTH_ATTACHMENT: Int
val STENCIL_ATTACHMENT: Int
val DEPTH_STENCIL_ATTACHMENT: Int
val NONE: Int
val FRAMEBUFFER_COMPLETE: Int
val FRAMEBUFFER_INCOMPLETE_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT: Int
val FRAMEBUFFER_INCOMPLETE_DIMENSIONS: Int
val FRAMEBUFFER_UNSUPPORTED: Int
val FRAMEBUFFER_BINDING: Int
val VALID_FRAMEBUFFER_OPERATION: Int
val UNPACK_FLIP_Y_WEBGL: Int
val UNPACK_PREMULTIPLY_ALPHA_WEBGL: Int
val CONTEXT_LOST_WEBGL: Int
val UNPACK_COLORSPACE_CONVERSION_WEBGL: Int
val BROWSER_DEFAULT_WEBGL: Int

```kotlin
/**
 * Exposes the Javascript [WebGLContextEvent](https://developer.mozilla.org/en/docs/Web/API/WebGLContextEvent) to Kotlin
 */
```
public external open class WebGLContextEvent(type: String, eventInit: WebGLContextEventInit = definedExternally): Event {
    open val statusMessage: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface WebGLContextEventInit: EventInit {
        val statusMessage: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    }


    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["statusMessage"] = statusMessage
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface ArrayBufferView: BufferDataSource {
        val buffer: ArrayBuffer
        val byteOffset: Int
        val byteLength: Int
        fun subarray(start: Int, end: Int): ArrayBufferView
        fun set(array: ArrayBuffer, offset: Int): ArrayBufferView
    }

    public external class ArrayBuffer(length: Int) : BufferDataSource {
        open val byteLength: Int
        fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer
    }

    /**
     * Exposes the JavaScript [ArrayBuffer](https://developer.mozilla.org/en/docs/Web/API/ArrayBuffer) to Kotlin
     */
    public external open class ArrayBuffer(length: Int): BufferDataSource {
        open val byteLength: Int
        fun slice(begin: Int, end: Int = definedExternally): ArrayBuffer
    }

    /**
     * Exposes the JavaScript [ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin
     */
    public external interface ArrayBufferView: BufferDataSource {
        val buffer: ArrayBuffer
        val byteOffset: Int
        val byteLength: Int
        fun subarray(start: Int, end: Int): ArrayBufferView
        fun set(array: ArrayBuffer, offset: Int): ArrayBufferView
    }

    public external open class ArrayBufferView: BufferDataSource {
        val buffer: ArrayBuffer
        val byteOffset: Int
        val byteLength: Int
        fun subarray(start: Int, end: Int): ArrayBufferView
        fun set(array: ArrayBuffer, offset: Int): ArrayBufferView
    }

    /**
     * Exposes the JavaScript [Int8Array](https://developer.mozilla.org/en/docs/Web/API/Int8Array) to Kotlin
     */
    public external open class Int8Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Int8Array)
        constructor(array: Array<Byte>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Int8Array, offset: Int = definedExternally)
        fun set(array: Array<Byte>, offset: Int = definedExternally)
        fun subarray(start: Int, end: Int): Int8Array
    }

    /**
     * Exposes the JavaScript [Uint8Array](https://developer.mozilla.org/en/docs/Web/API/Uint8Array) to Kotlin
     */
    public external open class Uint8Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Uint8Array)
        constructor(array: Array<Byte>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Uint8Array, offset: Int = definedExternally)
        fun set(array: Array<Byte>, offset: Int = definedExternally)
        fun subarray(start: Int, end: Int): Uint8Array
    }

    /**
     * Exposes the JavaScript [Uint8ClampedArray](https://developer.mozilla.org/en/docs/Web/API/Uint8ClampedArray) to Kotlin
     */
    public external open class Uint8ClampedArray: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Uint8ClampedArray)
        constructor(array: Array<Byte>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Uint8ClampedArray, offset: Int = definedExternally)
        fun set(array: Array<Byte>, offset: Int = definedExternally)
        fun subarray(start: Int, end: Int): Uint8ClampedArray
    }

    /**
     * Exposes the JavaScript [Int16Array](https://developer.mozilla.org/en/docs/Web/API/Int16Array) to Kotlin
     */
    public external open class Int16Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Int16Array)
        constructor(array: Array<Short>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Int16Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [Int32Array](https://developer.mozilla.org/en/docs/Web/API/Int32Array) to Kotlin
     */
    public external open class Int32Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Int32Array)
        constructor(array: Array<Short>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Int32Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [Uint16Array](https://developer.mozilla.org/en/docs/Web/API/Uint16Array) to Kotlin
     */
    public external open class Uint16Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Uint16Array)
        constructor(array: Array<Short>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Uint16Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [Uint32Array](https://developer.mozilla.org/en/docs/Web/API/Uint32Array) to Kotlin
     */
    public external open class Uint32Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Uint32Array)
        constructor(array: Array<Short>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Uint32Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [BigIntArray](https://developer.mozilla.org/en/docs/Web/API/BigIntArray) to Kotlin
     */
    public external open class BigIntArray: ArrayBufferView {
        constructor(length: Int)
        constructor(array: BigIntArray)
        constructor(array: Array<Short>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: BigIntArray, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [Float32Array](https://developer.mozilla.org/en/docs/Web/API/Float32Array) to Kotlin
     */
    public external open class Float32Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Float32Array)
        constructor(array: Array<Float>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Float32Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [Float64Array](https://developer.mozilla.org/en/docs/Web/API/Float64Array) to Kotlin
     */
    public external open class Float64Array: ArrayBufferView {
        constructor(length: Int)
        constructor(array: Float64Array)
        constructor(array: Array<Double>)
        constructor(buffer: ArrayBuffer, byteOffset: Int = definedExternally, length: Int = definedExternally)
        open val length: Int
        override val buffer: ArrayBuffer
        override val byteOffset: Int
        override val byteLength: Int
        fun set(array: Float64Array, offset: Int = definedExternally)
    }

    /**
     * Exposes the JavaScript [ArrayBufferView](https://developer.mozilla.org/en/docs/Web/API/ArrayBufferView) to Kotlin
     */
    public external open class ArrayBufferView: BufferDataSource {
        val buffer: ArrayBuffer
        val byteOffset: Int
        val byteLength: Int
    }

null
Open Source Used In IOx-FogDirector-AC18 1.17.0
org.w3c.dom.clipboard.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

public external abstract class MediaList : ItemArrayLike<String> {
    open var mediaText: String
    fun appendMedium(medium: String)
    fun deleteMedium(medium: String)
    override fun item(index: Int): String? = asDynamic()[index][**

@kotlin.internal.InlineOnly
public inline operator fun MediaList.get(index: Int): String? = asDynamic()[index][**

/**
 * Exposes the JavaScript
 */
public external abstract class StyleSheet {
    open val type: String
    open val href: String?
    open val ownerNode: UnionElementOrProcessingInstruction?
    open val parentStyleSheet: StyleSheet?
    open val title: String?
    open val media: MediaList
    open var disabled: Boolean
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSSStyleSheet : StyleSheet {
    open val ownerRule: CSSRule?
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)
}

/**
 * Exposes the JavaScript
 */
public external abstract class StyleSheetList : ItemArrayLike<StyleSheet> {
    override fun item(index: Int): StyleSheet?
}

@kotlin.internal.InlineOnly
public inline operator fun StyleSheetList.get(index: Int): StyleSheet? = asDynamic()[index][**

/**
 * Exposes the JavaScript
 */
public external interface LinkStyle {
    val sheet: StyleSheet?
        get() = definedExternally
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSSRuleList : ItemArrayLike<CSSRule> {
    override fun item(index: Int): CSSRule?
}

@kotlin.internal.InlineOnly
public inline operator fun CSSRuleList.get(index: Int): CSSRule? = asDynamic()[index][**

/**
 * Exposes the JavaScript
 */
public external abstract class CSSRule {
    open val type: Short
    open var cssText: String
    open val parentRule: CSSRule?
    open val parentStyleSheet: CSSStyleSheet?

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSSStyleRule : CSSRule {
    open var selectorText: String
    open val style: CSSStyleDeclaration

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

public external abstract class CSSImportRule : CSSRule {
    open val href: String
    open val media: MediaList
    open val styleSheet: CSSStyleSheet

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSSGroupingRule : CSSRule {
    open val cssRules: CSSRuleList
    fun insertRule(rule: String, index: Int): Int
    fun deleteRule(index: Int)

    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSSStyleDeclaration {
    companion object {
        val STYLE_RULE: Short
        val CHARSET_RULE: Short
        val IMPORT_RULE: Short
        val MEDIA_RULE: Short
        val FONT_FACE_RULE: Short
        val PAGE_RULE: Short
        val MARGIN_RULE: Short
        val NAMESPACE_RULE: Short
    }
}

open source used in IOx-FogDirector-AC18 1.17.0 2364
abstract class CSSMediaRule : CSSGroupingRule {
val media: MediaList

companion object {
val MEDIA_RULE: Short
val CHARSET_RULE: Short
val IMPORT_RULE: Short
val PAGE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}

/**
 * Exposes the JavaScript
[CSSPageRule](https://developer.mozilla.org/en/docs/Web/API/CSSPageRule) to Kotlin
 */
public external abstract class CSSPageRule : CSSGroupingRule {
val style: CSSStyleDeclaration

companion object {
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}

/**
 * Exposes the JavaScript
[CSSNamespaceRule](https://developer.mozilla.org/en/docs/Web/API/CSSNamespaceRule) to Kotlin
 */
public external abstract class CSSNamespaceRule : CSSRule {
val namespaceURI: String
val prefix: String

companion object {
val MEDIA_RULE: Short
val FONT_FACE_RULE: Short
val MARGIN_RULE: Short
val NAMESPACE_RULE: Short
}

/**
 * Exposes the JavaScript
[CSSStyleDeclaration](https://developer.mozilla.org/en/docs/Web/API/CSSStyleDeclaration) to Kotlin
 */
public external abstract class CSSStyleDeclaration : ItemArrayLike<String> {
val cssText: String
val parentRule: CSSRule?
val cssFloat: String
val alignContent: String
val alignItems: String
val alignSelf: String
val animation: String
val animationDelay: String
val animationDirection: String
val animationDuration: String
val animationFillMode: String
val animationIterationCount: String
val animationName: String
val animationPlayState: String
val animationTimingFunction: String
val backfaceVisibility: String
val background: String
val backgroundAttachment: String
val backgroundClip: String
val backgroundColor: String
val backgroundImage: String
val backgroundOrigin: String
val backgroundPosition: String
val backgroundRepeat: String
val backgroundSize: String
val borderBottom: String
val borderBottomColor: String
val borderBottomLeftRadius: String
val borderBottomRightRadius: String
val borderBottomStyle: String
val borderBottomWidth: String
val borderCollapse: String
val borderColor: String
val borderImage: String
val borderImageOutset: String
val borderImageRepeat: String
val borderImageSource: String
val borderImageWidth: String
val borderLeft: String
val borderLeftColor: String
val borderLeftStyle: String
val borderLeftWidth: String
val borderRadius: String
val borderRight: String
val borderRightColor: String
val borderRightStyle: String
val borderRightWidth: String
val borderTop: String
val borderTopColor: String
val borderTopLeftRadius: String
val borderTopRightRadius: String
val borderTopStyle: String
val borderTopWidth: String
val borderSpacing: String
val borderTop: String
val boxDecorationBreak: String
val boxShadow: String
val boxSizing: String
val breakAfter: String
val breakBefore: String
val breakInside: String
val captionSide: String
val clear: String
val clip: String
val color: String
val columnCount: String
val columnFill: String
val columnGap: String
val columnRule: String
val columnRuleColor: String
val columnRuleStyle: String
val columnRuleWidth: String
val columns: String
val content: String
val counterIncrement: String
}
String

open var counterReset: String
open var cursor: String
open var direction: String
open var display: String
open var emptyCells: String
open var filter: String
open var flex: String
open var flexBasis: String
open var flexDirection: String
open var flexFlow: String
open var flexGrow: String
open var flexShrink: String
open var flexWrap: String
open var font: String
open var fontFamily: String
open var fontFeatureSettings: String
open var fontKerning: String
open var fontWeight: String
open var fontSize: String
open var fontSizeAdjust: String
open var fontStyle: String
open var fontSynthesis: String
open var fontVariant: String
open var fontVariantAlternates: String
open var fontVariantCaps: String
open var fontVariantEastAsian: String
open var fontVariantLigatures: String
open var fontVariantNumeric: String
open var fontVariantPosition: String
open var fontWeight: String
open var flexGrow: String
open var flexShrink: String
open var flexGrow: String
open var flexShrink: String
open var justifyContent: String
open var letterSpacing: String
open var lineHeight: String
open var listStyle: String
open var listStyleImage: String
open var listStylePosition: String
open var listStyleType: String
open var margin: String
open var marginBottom: String
open var marginLeft: String
open var marginRight: String
open var marginTop: String
open var marginBottom: String
open var marginRight: String
open var marginTop: String
open var maxHeight: String
open var maxWidth: String
open var minHeight: String
open var minWidth: String
open var navDown: String
open var navIndex: String
open var navLeft: String
open var navRight: String
open var navUp: String
open var objectFit: String
open var objectPosition: String
open var opacity: String
open var order: String
open var orphans: String
open var outline: String
open var outlineColor: String
open var outlineOffset: String
open var outlineStyle: String
open var outlineWidth: String
open var overflowWrap: String
open var overflowX: String
open var overflowY: String
open var padding: String
open var paddingBottom: String
open var paddingLeft: String
open var paddingRight: String
open var paddingTop: String
open var pageBreakAfter: String
open var pageBreakBefore: String
open var pageBreakInside: String
open var perspective: String
open var perspectiveOrigin: String
open var phonemes: String
open var position: String
open var quotes: String
open var rest: String
open var restAfter: String
open var restBefore: String
open var right: String
open var tabSize: String
open var textAlign: String
open var textAlignmentLast: String
open var textCombineUpright: String
open var textDecoration: String
open var textDecorationColor: String
open var textDecorationLine: String
open var textDecorationStyle: String
open var textIndent: String
open var textJustify: String
open var textOrientation: String
open var textOverflow: String
open var textShadow: String
open var textTransform: String
open var textUnderlinePosition: String
open var top: String
open var transform: String
open var transformOrigin: String
open var transition: String
open var transitionDelay: String
open var transitionDuration: String
open var transitionProperty: String
open var transitionTimingFunction: String
open var unicodeBidi: String
open var verticalAlign: String
open var visibility: String
open var voiceBalance: String
open var voiceDuration: String
open var voicePitch: String
open var voicePitchRange: String
open var voiceRate: String
open var voiceStress: String
open var voiceVolume: String
open var whiteSpace: String
open var widows: String
open var wordBreak: String
open var wordSpacing: String
open var writingMode: String
open var zIndex: String
open var _dashed_attribute: String
open var _camel_cased_attribute: String
open var _webkit_cased_attribute: String
fun getPropertyValue(property: String): String
fun getPropertyPriority(property: String): String
fun setProperty(property: String, value: String, priority: String = definedExternally)
fun setPropertyValue(property: String, value: String)
fun setPropertyPriority(property: String, priority: String)
fun removeProperty(property: String): String
override fun item(index: Int):
String

@kotlin.internal.InlineOnly
public inline operator fun CSSStyleDeclaration.get(index: Int): String? =
asDynamic()[index]@/n

UnionElementOrProcessingInstruction",*/n * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming
Language contributors.*/n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.*/n * Note: This file is auto-generated, do not edit.*/n * see

github.com/kotlin/dukat for details.*/n

@kotlin.internal.InlineOnly
public inline operator fun CSSStyleDeclaration.get(index: Int): String? =
asDynamic()[index]

public external interface ElementCSSInlineStyle {
    val style:
        CSSStyleDeclaration
}

/**
 * Exposes the JavaScript
 */
public external abstract class CSS {
    companion object {
        fun escape(ident: String): String
    }
}

public external interface

Open Source Used In IOx-FogDirector-AC18 1.17.0  2367
definedExternally
    set(value) = definedExternally
    var buttons: Short? /* = 0 */
    get() =

definedExternally
    set(value) = definedExternally
    var relatedTarget: EventTarget? /* = null */
    get() =

definedExternally
    set(value) = definedExternally
    var region: String? /* = null */
    get() =
definedExternally


MouseEventInit
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey;
    o["shiftKey"] = shiftKey;
    o["altKey"] = altKey;
    o["metaKey"] = metaKey;
    o["modifierAltGraph"] = modifierAltGraph;
    o["modifierCapsLock"] = modifierCapsLock;
    o["modifierFn"] = modifierFn;
    o["modifierFnLock"] = modifierFnLock;
    o["modifierHyper"] = modifierHyper;
    o["modifierNumLock"] = modifierNumLock;
    o["modifierScrollLock"] = modifierScrollLock;
    o["modifierSuper"] = modifierSuper;
    o["modifierSymbol"] = modifierSymbol;
    o["modifierSymbolLock"] = modifierSymbolLock;
    o["view"] = view;
    o["detail"] = detail;
    o["bubbles"] = bubbles;
    o["cancelable"] = cancelable;
    o["composed"] = composed;
    return o
}

EventModifierInit { val o = js("({})")
    o["ctrlKey"] = ctrlKey;
    o["shiftKey"] = shiftKey;
    o["altKey"] = altKey;
    o["metaKey"] = metaKey;
    o["modifierAltGraph"] = modifierAltGraph;
    o["modifierCapsLock"] = modifierCapsLock;
    o["modifierFn"] = modifierFn;
    o["modifierFnLock"] = modifierFnLock;
    o["modifierHyper"] = modifierHyper;
    o["modifierNumLock"] = modifierNumLock;
    o["modifierScrollLock"] = modifierScrollLock;
    o["modifierSuper"] = modifierSuper;
    o["modifierSymbol"] = modifierSymbol;
    o["modifierSymbolLock"] = modifierSymbolLock;
    o["view"] = view;
    o["detail"] = detail;
    o["bubbles"] = bubbles;
    o["cancelable"] = cancelable;
    o["composed"] = composed;
    return o
}
```kotlin
/**
 * Exposes the JavaScript
 [WheelEvent](https://developer.mozilla.org/en/docs/Web/API/WheelEvent) to Kotlin
 */
public external open class WheelEvent(type: String, eventInitDict: WheelEventInit = definedExternally) : MouseEvent {
    open val deltaX: Double
    open val deltaY: Double
    open val deltaZ: Double
    open val deltaMode: Int

    companion object {
        val DOM_DELTA_PIXEL: Int
        val DOM_DELTA_LINE: Int
        val DOM_DELTA_PAGE: Int
    }
}

public external interface WheelEventInit : MouseEventInit {
    var deltaX: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaY: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaZ: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var deltaMode: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["deltaX"] = deltaX
    o["deltaY"] = deltaY
    o["deltaZ"] = deltaZ
    o["deltaMode"] = deltaMode
    o["screenX"] = screenX
    o["screenY"] = screenY
    o["clientX"] = clientX
    o["clientY"] = clientY
    o["button"] = button
    o["buttons"] = buttons
    o["relatedTarget"] = relatedTarget
    o["region"] = region
    o["ctrlKey"] = ctrlKey
    o["shiftKey"] = shiftKey
    o["altKey"] = altKey
    o["metaKey"] = metaKey
    o["modifierAltGraph"] = modifierAltGraph
    o["modifierCapsLock"] = modifierCapsLock
    o["modifierFn"] = modifierFn
    o["modifierFnLock"] = modifierFnLock
    o["modifierHyper"] = modifierHyper
    o["modifierNumLock"] = modifierNumLock
    o["modifierScrollLock"] = modifierScrollLock
    o["modifierSuper"] = modifierSuper
    o["modifierSymbol"] = modifierSymbol
    o["modifierSymbolLock"] = modifierSymbolLock
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [InputEvent](https://developer.mozilla.org/en/docs/Web/API/InputEvent) to Kotlin
 */
public external open class InputEvent(type: String, eventInitDict: InputEventInit = definedExternally) : UIEvent {
    open val data: String
    open var isComposing: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface InputEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["isComposing"] = isComposing
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [KeyboardEvent](https://developer.mozilla.org/en/docs/Web/API/KeyboardEvent) to Kotlin
 */
public external open class KeyboardEvent(type: String, eventInitDict: KeyboardEventInit = definedExternally) : UIEvent {
    open val key: String
    open val code: String
    open val location: Int
    open val ctrlKey: Boolean
    open val shiftKey: Boolean
    open val altKey: Boolean
    open val metaKey: Boolean

    companion object {
        val CANCOMPOSE: Boolean
        val CANCANCEL: Boolean
        val CANCANCELABLE: Boolean
        val CANCOMPOSEABLE: Boolean
    }
}
```

Open Source Used in IOX-FogDirector-AC18 1.1.7.0 2369
DOM_KEY_LOCATION_STANDARD: Int
DOM_KEY_LOCATION_LEFT: Int
DOM_KEY_LOCATION_RIGHT: Int
DOM_KEY_LOCATION_NUMPAD: Int

Short
val CAPTURING_PHASE: Short
val AT_TARGET: Short
val BUBBLING_PHASE: Short

unpublished external interface KeyboardEventInit: EventModifierInit {

var key: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
    var code: String? /* = "" */
    get() = definedExternally
    set(value) = definedExternally
    var location: Int? /* = 0 */
    get() = definedExternally
    set(value) = definedExternally
    var repeat: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var isComposing: Boolean? /* = false */
    get() = definedExternally
    set(value) = definedExternally
    var ctrlKey: Boolean? = false
    var shiftKey: Boolean? = false
    var altKey: Boolean? = false
    var metaKey: Boolean? = false
    var modifierAltGraph: Boolean? = false
    var modifierCapsLock: Boolean? = false
    var modifierFn: Boolean? = false
    var modifierNumLock: Boolean? = false
    var modifierScrollLock: Boolean? = false
    var modifierSuper: Boolean? = false
    var modifierSymbol: Boolean? = false
    var modifierSymbolLock: Boolean? = false
    var view: Window? = null
    var detail: Int? = 0
    var bubbles: Boolean? = false
    var cancelable: Boolean? = false
    var composed: Boolean? = false

    initEvent(type: String, bubbles: Boolean, cancelable: Boolean)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface KeyboardEventInit : EventModifierInit {
    var key: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var code: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var location: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var repeat: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var isComposing: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ctrlKey: Boolean? = false
    var shiftKey: Boolean? = false
    var altKey: Boolean? = false
    var metaKey: Boolean? = false
    var modifierAltGraph: Boolean? = false
    var modifierCapsLock: Boolean? = false
    var modifierFn: Boolean? = false
    var modifierNumLock: Boolean? = false
    var modifierScrollLock: Boolean? = false
    var modifierSuper: Boolean? = false
    var modifierSymbol: Boolean? = false
    var modifierSymbolLock: Boolean? = false
    var view: Window? = null
    var detail: Int? = 0
    var bubbles: Boolean? = false
    var cancelable: Boolean? = false
    var composed: Boolean? = false

        isComposing: Boolean? = false, ctrlKey: Boolean? = false, shiftKey: Boolean? = false, altKey: Boolean? = false,
        metaKey: Boolean? = false, modifierAltGraph: Boolean? = false, modifierCapsLock: Boolean? = false,
        modifierFn: Boolean? = false, modifierNumLock: Boolean? = false, modifierScrollLock: Boolean? = false,
        composed: Boolean? = false): KeyboardEventInit {
        val o = js("({})")
        o["key"] = key
        o["code"] = code
        o["location"] = location
        o["repeat"] = repeat
        o["isComposing"] = isComposing
        o["ctrlKey"] = ctrlKey
        o["shiftKey"] = shiftKey
        o["altKey"] = altKey
        o["metaKey"] = metaKey
        o["modifierAltGraph"] = modifierAltGraph
        o["modifierCapsLock"] = modifierCapsLock
        o["modifierFn"] = modifierFn
        o["modifierNumLock"] = modifierNumLock
        o["modifierScrollLock"] = modifierScrollLock
        o["modifierSuper"] = modifierSuper
        o["modifierSymbol"] = modifierSymbol
        o["modifierSymbolLock"] = modifierSymbolLock
        o["view"] = view
        o["detail"] = detail
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external open class CompositionEvent(type: String, eventInitDict: CompositionEventInit = definedExternally) : UIEvent {
    open val data: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface CompositionEventInit : UIEventInit {
    var data: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CompositionEventInit(data: String? = "", view: Window? = null, detail: Int? = 0, bubbles: Boolean? = false,
    cancelable: Boolean? = false, composed: Boolean? = false): CompositionEventInit {
    val o = js("({})")
    o["data"] = data
    o["view"] = view
    o["detail"] = detail
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [CompositionEvent](https://developer.mozilla.org/en/docs/Web/API/CompositionEvent) to Kotlin
 */
public external open class Event(type: String, eventInitDict: EventInit = definedExternally) {
    open val type: String
    open val target: EventTarget?
    open val currentTarget: EventTarget?
    open val eventPhase: Short
    open val defaultPrevented: Boolean
    open val isTrusted: Boolean
    open val timeStamp: Number
    fun composedPath(): Array<EventTarget>
    fun stopPropagation(): Unit
    fun stopImmediatePropagation(): Unit
    fun preventDefault(): Unit
    fun initEvent(type: String, bubbles: Boolean, cancelable: Boolean) {
        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }
    }

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

/**
 * Exposes the JavaScript [Event](https://developer.mozilla.org/en/docs/Web/API/Event) to Kotlin
 */
public external abstract class EventTarget {
    fun addEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)
    fun addEventListener(type: String, callback: (Event) -> Unit?, options: dynamic = definedExternally)
    fun removeEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)
    fun addEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)
    fun removeEventListener(type: String, callback: EventListener?, options: dynamic = definedExternally)

public external abstract class HTMLUnknownElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [DOMStringMap](https://developer.mozilla.org/en/docs/Web/API/DOMStringMap) to Kotlin
 */
public external abstract class DOMStringMap

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.get(name: String): String? = asDynamic()[name]

@kotlin.internal.InlineOnly
public inline operator fun DOMStringMap.set(name: String, value: String) { asDynamic()[name] = value }

/**
 * Exposes the JavaScript [HTMLHtmlElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHtmlElement) to Kotlin
 */
public external abstract class HTMLHtmlElement : HTMLElement {
    open var version: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHeadElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadElement) to Kotlin
 */
public external abstract class HTMLHeadElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTitleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTitleElement) to Kotlin
 */
public external abstract class HTMLTitleElement : HTMLElement {
    open var text: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLBaseElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBaseElement) to Kotlin
 */
public external abstract class HTMLBaseElement : HTMLElement {
external abstract class HTMLBaseElement : HTMLElement {
    open var href: String
    open var target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLinkElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLinkElement) to Kotlin
 */
public external abstract class HTMLLinkElement : HTMLElement, LinkStyle {
    open var href: String
    open var crossOrigin: String?
    open var rel: String
    open var `as`: RequestDestination
    open val relList: DOMTokenList
    open var media: String
    open var nonce: String
    open var hreflang: String
    open var type: String
    open var sizes: DOMTokenList
    open var referrerPolicy: String
    open var charset: String
    open var rev: String
    open var target: String
    open var scope: String
    open var workerType: WorkerType

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMetaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMetaElement) to Kotlin
 */
public external abstract class HTMLMetaElement : HTMLElement {
    open var name: String
    open var httpEquiv: String
    open var content: String
    open var scheme: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLStyleElement](https://developer.mozilla.org/en/docs/Web/API/HTMLStyleElement) to Kotlin
 */
public external abstract class HTMLStyleElement : HTMLElement, LinkStyle {
    open var media: String
    open var nonce: String
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_DISCONNECTED: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short

val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLIMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENTPOSITION_IMPLEMENTATION_SPECIFIC: Short

val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENTPOSITION_IMPLEMENTATION_SPECIFIC: Short

/* Exposes the JavaScript [HTMLBodyElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBodyElement) to Kotlin */

public external abstract class HTMLBodyElement : HTMLElement, WindowEventHandlers {
    open var text: String
    open var link: String
    open var aLink: String
    open var bgColor: String
    open var background: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript [HTMLHeadingElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHeadingElement) to Kotlin */

public external abstract class HTMLHeadingElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript [HTMLParagraphElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParagraphElement) to Kotlin */

public external abstract class HTMLParagraphElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/* Exposes the JavaScript [HTMLHRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLHRElement) to Kotlin */

public external abstract class HTMLHRElement : HTMLElement {
    open var align: String
    open var color: String
    open var noShade: Boolean
    open var size: String
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
/**
 * Exposes the JavaScript HTMLPreElement to Kotlin
 */
public external abstract class HTMLPreElement : HTMLElement {
    open var width: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLQuoteElement to Kotlin
 */
public external abstract class HTMLQuoteElement : HTMLElement {
    open var cite: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLOListElement to Kotlin
 */
public external abstract class HTMLOListElement : HTMLElement {
    open var reversed: Boolean
    open var start: Int
    open var type: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLUListElement to Kotlin
 */
public external abstract class HTMLUListElement : HTMLElement {
    open var compact: Boolean
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript HTMLLIElement to Kotlin
 */
public external abstract class HTMLLIElement : HTMLElement {
    open var value: Int
    open var type: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLDListElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDListElement) to Kotlin
 */
public external abstract class HTMLDListElement : HTMLElement {
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDivElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDivElement) to Kotlin
 */
public external abstract class HTMLDivElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAnchorElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAnchorElement) to Kotlin
 */
public external abstract class HTMLAnchorElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open val relList: DOMTokenList
    open var hreflang: String
    open var type: String
    open var text: String
    open var referrerPolicy: String
    open var coords: String
    open var charset: String
    open var name: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLDataElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDataElement) to Kotlin
 */
public external abstract class HTMLDataElement : HTMLElement {
    open var value: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTimeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTimeElement) to Kotlin
 */
public external abstract class HTMLTimeElement : HTMLElement {
    open var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSpanElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSpanElement) to Kotlin
 */
public external abstract class HTMLSpanElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLBRElement](https://developer.mozilla.org/en/docs/Web/API/HTMLBRElement) to Kotlin
 */
public external abstract class HTMLBRElement : HTMLElement {
    open var clear: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLHyperlinkElementUtils](https://developer.mozilla.org/en/docs/Web/API/HTMLHyperlinkElementUtils) to Kotlin
 */
public external interface HTMLHyperlinkElementUtils {
    var href: String
    val origin: String
    var protocol: String
    var username: String
    var password: String
    var host: String
    var hostname: String
    var port: String
    var pathname: String
    var search: String
    var hash: String
}

/**
 * Exposes the JavaScript [HTMLModElement](https://developer.mozilla.org/en/docs/Web/API/HTMLModElement) to Kotlin
 */
public external abstract class HTMLModElement : HTMLElement {
    open var cite: String
    open
var dateTime: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLPictureElement](https://developer.mozilla.org/en/docs/Web/API/HTMLPictureElement) to Kotlin
 */
public external abstract class HTMLPictureElement : HTMLElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSourceElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSourceElement) to Kotlin
 */
public external abstract class HTMLSourceElement : HTMLElement {
    open var src: String
    open var type: String
    open var srcset: String
    open var sizes: String
    open var media: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLImageElement](https://developer.mozilla.org/en/docs/Web/API/HTMLImageElement) to Kotlin
 */
public external abstract class HTMLImageElement : HTMLElement, HTMLOrSVGImageElement, TexImageSource {
    open var alt: String
    open var src: String
    open var srcset: String
    open var sizes: String
    open var crossOrigin: String?
    open var useMap: String
    open var isMap: Boolean
    open var width: Int
    open var height: Int
    open val naturalWidth: Int
    open val naturalHeight: Int
    open val complete: Boolean
    open var lowsrc: String
    open var align: String
    open var hspace: Int
    open var vspace: Int
    open var longDesc: String
    open var border: String
    open val x: Int
    open val y: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
*/

/**
 * Exposes the JavaScript [HTMLIFrameElement](https://developer.mozilla.org/en/docs/Web/API/HTMLIFrameElement) to Kotlin
 */
public external abstract class HTMLIFrameElement : HTMLElement {
    open var src: String
    open var srcdoc: String
    open var name: String
    open val sandbox: DOMTokenList
    open var allowfullscreen: Boolean
    open var allowusermedia: Boolean
    open var width: String
    open var height: String
    open var referrerpolicy: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var align: String
    open var scrolling: String
    open var frameborder: String
    open var longdesc: String
    open var marginheight: String
    open var marginwidth: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLEmbedElement](https://developer.mozilla.org/en/docs/Web/API/HTMLEmbedElement) to Kotlin
 */
public external abstract class HTMLEmbedElement : HTMLElement {
    open var src: String
    open var type: String
    open var width: String
    open var height: String
    open var align: String
    open var name: String
    fun getSVGDocument(): Document?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLObjectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLObjectElement) to Kotlin
 */
public external abstract class HTMLObjectElement : HTMLElement {
    open var data: String
    open var type: String
    open var typeMustMatch: Boolean
    open var name: String
    open var useMap: String
    open var form: HTMLFormElement?
    open var width: String
    open var height: String
    open var contentDocument: Document?
    open var contentWindow: Window?
    open var willValidate: Boolean
    open var validity: ValidityState
    open var validationMessage: String
    open var align: String
    open var archive: String
    open var standby: String
    open var codeBase: String
    open var codeType: String
    open var border: String
    fun getSVGDocument(): Document?
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2379
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/* Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin */

/**
 * Exposes the JavaScript [HTMLParamElement](https://developer.mozilla.org/en/docs/Web/API/HTMLParamElement) to Kotlin */

public abstract class HTMLParamElement : HTMLElement {
    open var name: String
    open var value: String
    open var type: String
    open var valueType: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLVideoElement](https://developer.mozilla.org/en/docs/Web/API/HTMLVideoElement) to Kotlin */

public abstract class HTMLVideoElement : HTMLMediaElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    open val videoWidth: Int
    open val videoHeight: Int
    open var poster: String
    open var playsInline: Boolean

    companion object {
        val NETWORK_LOADED: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLAudioElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAudioElement) to Kotlin */

public abstract class HTMLAudioElement : HTMLMediaElement {
    companion object {
        val NETWORK_LOADED: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTrackElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTrackElement) to Kotlin */

public abstract class HTMLTrackElement : HTMLElement {
    open var kind: String
    open var src: String
    open var srclang: String
    open var label: String
    open var default: Boolean
    open val readyState: Short
    open val track: TextTrack

    companion object {
        val NONE: Short
        val LOADING: Short
        valTRANSCODED: Short
        val PARSED: Short
        val FINISHED: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}
**Exposes the JavaScript \[HTMLMediaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMediaElement) to Kotlin**

```kotlin
public external abstract class HTMLMediaElement : HTMLElement {
    open val error: MediaError?
    open var src: String
    open var srcObject: dynamic
    open val currentSrc: String
    open var crossOrigin: String?
    open val networkState: Short
    open var preload: String
    open val buffered: TimeRanges
    open val readystatechange: Short
    open var seekable: TimeRanges
    open val ended: Boolean
    open var autoplay: Boolean
    open var loop: Boolean
    open var controls: Boolean
    open var volume: Double
    open var muted: Boolean
    open var defaultMuted: Boolean
    open val audioTracks: AudioTrackList
    open val videoTracks: VideoTrackList
    open val textTracks: TextTrackList
    fun load()
    fun canPlayType(type: String): CanPlayTypeResult
    fun fastSeek(time: Double)
    fun getStartDate(): dynamic
    fun play(): Promise<Unit>
    fun pause()
    fun addTextTrack(kind: TextTrackKind, label: String = definedExternally, language: String = definedExternally): TextTrack
}
```

**Exposes the JavaScript \[MediaError](https://developer.mozilla.org/en/docs/Web/API/MediaError) to Kotlin**

```kotlin
public external abstract class MediaError {
    open val code: Short
}
```

**Exposes the JavaScript \[AudioTrackList](https://developer.mozilla.org/en/docs/Web/API/AudioTrackList) to Kotlin**

```kotlin
public external abstract class AudioTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): AudioTrack?
}
```

**Exposes the JavaScript \[AudioTrack](https://developer.mozilla.org/en/docs/Web/API/AudioTrack) to Kotlin**

```kotlin
public external abstract class AudioTrack : UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var enabled: Boolean
}
```

**Exposes the JavaScript \[VideoTrackList](https://developer.mozilla.org/en/docs/Web/API/VideoTrackList) to Kotlin**

```kotlin
}`
```
public external abstract class VideoTrackList : EventTarget {
    open val length: Int
    open val selectedIndex: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): VideoTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun VideoTrackList.get(index: Int): VideoTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [VideoTrack](https://developer.mozilla.org/en/docs/Web/API/VideoTrack) to Kotlin
 */
public external abstract class VideoTrack : UnionAudioTrackOrTextTrackOrVideoTrack {
    open val id: String
    open val kind: String
    open val label: String
    open val language: String
    open var selected: Boolean
}

public external abstract class TextTrackList : EventTarget {
    open val length: Int
    open var onchange: ((Event) -> dynamic)?
    open var onaddtrack: ((TrackEvent) -> dynamic)?
    open var onremovetrack: ((TrackEvent) -> dynamic)?
    fun getTrackById(id: String): TextTrack?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackList.get(index: Int): TextTrack? = asDynamic()[index]

/**
 * Exposes the JavaScript [TextTrack](https://developer.mozilla.org/en/docs/Web/API/TextTrack) to Kotlin
 */
public external abstract class TextTrack : EventTarget, UnionAudioTrackOrTextTrackOrVideoTrack {
    open val kind: TextTrackKind
    open val label: String
    open val language: String
    open val id: String
    open val inBandMetadataTrackDispatchType: String
    open var mode: TextTrackMode
    open val cues: TextTrackCueList?
    open val activeCues: TextTrackCueList?
    open var oncuechange: ((Event) -> dynamic)?
    fun addCue(cue: TextTrackCue)
    fun removeCue(cue: TextTrackCue)
}

public external abstract class TextTrackCueList {
    open val length: Int
    fun getCueById(id: String): TextTrackCue?
}

@kotlin.internal.InlineOnly
public inline operator fun TextTrackCueList.get(index: Int): TextTrackCue? = asDynamic()[index]

/**
 * Exposes the JavaScript [TextTrackCue](https://developer.mozilla.org/en/docs/Web/API/TextTrackCue) to Kotlin
 */
public external abstract class TextTrackCue : EventTarget {
    open val track: TextTrack?
    open var id: String
    open var startTime: Double
    open var endTime: Double
    open var pauseOnExit: Boolean
    open var onenter: ((Event) -> dynamic)?
    open var onexit: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript [TimeRanges](https://developer.mozilla.org/en/docs/Web/API/TimeRanges) to Kotlin
 */
public external abstract class TimeRanges {
    open val length: Int
    fun start(index: Int): Double
    fun end(index: Int): Double
}

/**
 * Exposes the JavaScript [TrackEvent](https://developer.mozilla.org/en/docs/Web/API/TrackEvent) to Kotlin
 */
public external open class TrackEvent(type: String, eventInitDict: TrackEventInit = definedExternally) : Event {
    open val track: UnionAudioTrackOrTextTrackOrVideoTrack?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface TrackEventInit : EventInit {
    var track: UnionAudioTrackOrTextTrackOrVideoTrack? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var value: Any
    var newValue: Any
    var oldValue: Any
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["track"] = track
    o["value"] = value
    o["newValue"] = newValue
    o["oldValue"] = oldValue
    return o
}

/**
 * Exposes the JavaScript [HTMLMapElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMapElement) to Kotlin
 */
public external abstract class HTMLMapElement : HTMLElement {
    open var name: String
    open val areas: HTMLCollection

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_INFERIOR: Short
    }
}
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

Exposes the JavaScript [HTMLAreaElement](https://developer.mozilla.org/en/docs/Web/API/HTMLAreaElement) to Kotlin

public external abstract class HTMLAreaElement : HTMLElement, HTMLHyperlinkElementUtils {
    open var coords: String
    open var shape: String
    open var target: String
    open var download: String
    open var ping: String
    open var rel: String
    open var relList: DOMTokenList
    open var referrerPolicy: String
    open var noHref: Boolean
    companion object {
        val ELEMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableElement) to Kotlin
 */
public external abstract class HTMLTableElement : HTMLElement {
    open var caption: HTMLTableCaptionElement?
    open var tHead: HTMLTableSectionElement?
    open var tFoot: HTMLTableSectionElement?
    open val tBodies: HTMLCollection
    open val rows: HTMLCollection
    open val align: String
    open val border: String
    open val frame: String
    open val rules: String
    open var summary: String
    open var width: String
    open var bgColor: String
    open var cellPadding: String
    open var cellSpacing: String
    fun createCaption(): HTMLTableCaptionElement
    fun deleteCaption()
    fun createTHead(): HTMLTableSectionElement
    fun deleteTHead()
    fun createTFoot(): HTMLTableSectionElement
    fun insertRow(index: Int = definedExternally): HTMLTableRowElement
    fun deleteRow(index: Int)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableCaptionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCaptionElement) to Kotlin
 */
public external abstract class HTMLTableCaptionElement : HTMLElement {
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTableColElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableColElement) to Kotlin
 */
public external abstract class HTMLTableColElement : HTMLElement {
    open var span: Int

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
align: String
open var ch: String open var chOff: String open var vAlign: String open var width:
open var companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLTableSectionElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableSectionElement) to Kotlin
 */
public external abstract class HTMLTableSectionElement : HTMLElement {
open val rows: HTMLCollection
open var align: String
open var ch: String
open var chOff: String
open var vAlign: String

fun insertRow(index: Int = definedExternally): HTMLElement
fun deleteRow(index: Int)

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLTableRowElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableRowElement) to Kotlin
 */
public external abstract class HTMLTableRowElement : HTMLElement {
open val rowIndex: Int
open val sectionRowIndex: Int
open var cells: HTMLCollection
open var align: String
open var ch: String
open var chOff: String
open var vAlign: String
open var bgColor: String

fun insertCell(index: Int = definedExternally): HTMLElement
fun deleteCell(index: Int)

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript [HTMLTableCellElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTableCellElement) to Kotlin
 */
public external abstract class HTMLTableCellElement : HTMLElement {
open var colSpan: Int
open var rowSpan: Int
open var headers: String
open val cellIndex: Int
open var scope: String
open var abbr: String
open var align: String
open var axis: String
open var height: String
open var width: String
open var ch: String
open var chOff: String
open var noWrap: Boolean
open var vAlign: String
open var bgColor: String

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

DOCUMENT_TYPE_NODE: Short
DOCUMENT_FRAGMENT_NODE: Short
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript
[HTMLFormElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFormElement) to Kotlin
 */
public external abstract class HTMLFormElement : HTMLElement {
    open var acceptCharset: String
    open var action: String
    open var autocomplete: String
    open var enctype: String
    open var encoding: String
    open var method: String
    open var name: String
    open var noValidate: Boolean
    open var target: String
    open val elements: HTMLFormControlsCollection
    open val length: Int
    fun submit()
    fun reset()
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(index: Int): Element? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun HTMLFormElement.get(name: String): UnionElementOrRadioNodeList? = asDynamic()[name]

/**
 * Exposes the JavaScript
[HTMLLabelElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLabelElement) to Kotlin
 */
public external abstract class HTMLLabelElement : HTMLElement {
    open var htmlFor: String
    open val control: HTMLElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[HTMLInputElement](https://developer.mozilla.org/en/docs/Web/API/HTMLInputElement) to Kotlin
 */
public external abstract class HTMLInputElement : HTMLElement {
    open var accept: String
    open var alt: String
    open var autocomplete: String
    open var autofocus: Boolean
    open var defaultChecked: Boolean
    open var checked: Boolean
    open var dirName: String
    open var disabled: Boolean
    open var form: HTMLFormElement?
    open var files: FileList?
    open var formAction: String
    open var formEnctype: String
    open var formMethod: String
    open var formNoValidate: Boolean
    open var formTarget: String
    open var height: Int
    open var indeterminate: Boolean
    open var inputMode: String
    open var list: HTMLElement?
    open var max: String
    open var maxLength: Int
    open var min: String
    open var minLength: Int
    open var multiple: Boolean
    open var name: String
    open var pattern: String
    open var placeholder: String
    open var readOnly: Boolean
    open var required: Boolean
    open var size: Int
    open var step: String
    open var src: String
    open var type: String
    open var defaultValue: String
    open var value: String
    open var valueAsDate: dynamic
    open var valueAsNumber: Double
    open var width: Int
    open var willValidate: Boolean
    open var validationMessage: String
    open var labels: NodeList
    open var selectionStart: Int
    open var selectionEnd: Int
    open var val
selectionDirection: String?
open var align: String?
open var useMap: String?
fun stepUp(n: Int = definedExternally)
fun stepDown(n: Int = definedExternally)
fun checkValidity(): Boolean?
fun reportValidity(): Boolean?
fun setCustomValidity(error: String?)
fun select(i: Int = definedExternally)
fun setRangeText(replacement: String?)
fun setRangeText(replacement: String, start: Int, end: Int, selectionMode: SelectionMode = definedExternally)
fun setSelectionRange(start: Int, end: Int, direction: String = definedExternally)

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[HTMLButtonElement](https://developer.mozilla.org/en/docs/Web/API/HTMLButtonElement) to Kotlin
*/
public external abstract class HTMLButtonElement : HTMLElement {
open var autofocus: Boolean
open var disabled: Boolean
open val form: HTMLFormElement?
open var formAction: String
open var formEnctype: String
open var formMethod: String
open var formNoValidate: Boolean
open var formTarget: String
open var name: String
open var type: String
open var value: String
open var menu: HTMLMenuElement?
open var willValidate: Boolean
open var validity: ValidityState
open var validationMessage: String
open val labels: NodeList
fun checkValidity(): Boolean
fun reportValidity(): Boolean
fun setCustomValidity(error: String)

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val NOTATION_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[HTMLSelectElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSelectElement) to Kotlin
*/
public external abstract class HTMLSelectElement : HTMLElement, ItemArrayLike<Element> {
open var autocomplete: String
open var autofocus: Boolean
open var disabled: Boolean
open val form: HTMLFormElement?
open var multiple: Boolean
open var name: String
open var required: Boolean
open var size: Int
open var type: String
open val options: HTMLOptionsCollection
override var length: Int
open val selectedOptions: HTMLCollection
override var options: HTMLOptionsCollection
override var value: String
open var value: String
open var menu: HTMLMenuElement
open var willValidate: Boolean
open var validity: ValidityState
open var validationMessage: String
open val labels: NodeList
fun namedItem(name: String): HTMLOptionElement?
fun add(element: UnionHTMLOptGroupElementOrHTMLOptionElement, before: dynamic = definedExternally)
fun remove(index: Int)
fun checkValidity(): Boolean
fun reportValidity(): Boolean
fun setCustomValidity(error: String?)
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
fun override fun item(index: Int): Element?
```kotlin
/**
 * Exposes the JavaScript [HTMLProgressElement](https://developer.mozilla.org/en/docs/Web/API/HTMLProgressElement) to Kotlin
 */
public external abstract class HTMLProgressElement : HTMLElement {
    open var value: Double
    open var max: Double
    open val position: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMeterElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMeterElement) to Kotlin
 */
public external abstract class HTMLMeterElement : HTMLElement {
    open var value: Double
    open var min: Double
    open var max: Double
    open var low: Double
    open var high: Double
    open var optimum: Double
    open val labels: NodeList

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFieldSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFieldSetElement) to Kotlin
 */
public external abstract class HTMLFieldSetElement : HTMLElement {
    open var disabled: Boolean
    open val form: HTMLFormElement?
    open var name: String
    open var type: String
    open val elements: HTMLCollection
    open val willValidate: Boolean
    open val validity: ValidityState
    open val validationMessage: String
    fun checkValidity(): Boolean
    fun reportValidity(): Boolean
    fun setCustomValidity(error: String)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLLegendElement](https://developer.mozilla.org/en/docs/Web/API/HTMLLegendElement) to Kotlin
 */
public external abstract class HTMLLegendElement : HTMLElement {
    open var form: HTMLFormElement?
    open var align: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```

---

Open Source Used in IOX-FogDirector-AC18 1.17.0 2389
/**
 * Exposes the JavaScript [ValidityState](https://developer.mozilla.org/en/docs/Web/API/ValidityState) to Kotlin
 */
public external abstract class ValidityState {
    open val valueMissing: Boolean
    open val typeMismatch: Boolean
    open val patternMismatch: Boolean
    open val tooLong: Boolean
    open val tooShort: Boolean
    open val rangeUnderflow: Boolean
    open val rangeOverflow: Boolean
    open val stepMismatch: Boolean
    open val badInput: Boolean
    open val customError: Boolean
    open val valid: Boolean
}

/**
 * Exposes the JavaScript [HTMLDetailsElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDetailsElement) to Kotlin
 */
public external abstract class HTMLDetailsElement : HTMLElement {
    open var open: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLMenuElement : HTMLElement {
    open var type: String
    open var label: String
    open var compact: Boolean

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external open class RelatedEvent(type: String, eventInitDict: RelatedEventInit = definedExternally) : Event {
    open val relatedTarget: EventTarget?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface RelatedEventInit : EventInit {
    var relatedTarget: EventTarget? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun RelatedEventInit(relatedTarget:
RelatedEventInit {
    \n    o["relatedTarget"] = relatedTarget
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}\n** Exposes the JavaScript [HTMLDialogElement](https://developer.mozilla.org/en/docs/Web/API/HTMLDialogElement) to Kotlin

*Anpublic external abstract class HTMLDialogElement : HTMLElement {
    \n    open var open: Boolean
    open var returnValue: String
    fun show(anchor: UnionElementOrMouseEvent = definedExternally)
    fun showModal(anchor: UnionElementOrMouseEvent = definedExternally)
    fun close(returnValue: String = definedExternally)

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLScriptElement](https://developer.mozilla.org/en/docs/Web/API/HTMLScriptElement) to Kotlin
 */
public external abstract class HTMLScriptElement : HTMLElement, HTMLOrSVGScriptElement {
    open var src: String
    open var type: String
    open var charset: String
    open var async: Boolean
    open var defer: Boolean
    open var crossOrigin: String?
    open var text: String
    open var nonce: String
    open var event: String
    open var htmlFor: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLTemplateElement](https://developer.mozilla.org/en/docs/Web/API/HTMLTemplateElement) to Kotlin
 */
public external abstract class HTMLTemplateElement : HTMLElement {
    open var content: DocumentFragment

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [HTMLSlotElement](https://developer.mozilla.org/en/docs/Web/API/HTMLSlotElement) to Kotlin
 */
public external abstract class HTMLSlotElement : HTMLElement {
    open var name: String
    fun assignedNodes(options: AssignedNodesOptions = definedExternally): Array<Node>

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
    }
}

Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short

public external interface AssignedNodesOptions {
    var flatten: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun AssignedNodesOptions(flatten: Boolean? = false): AssignedNodesOptions {
    val o = js("({})")
    o["flatten"] = flatten
    return o
}

/**
 * Exposes the JavaScript [HTMLCanvasElement](https://developer.mozilla.org/en/docs/Web/API/HTMLCanvasElement) to Kotlin
 */
public external abstract class HTMLCanvasElement : HTMLElement, CanvasImageSource, TexImageSource {
    open var width: Int
    open var height: Int
    fun getContext(contextId: String, vararg arguments: Any?): RenderingContext?
    fun toDataURL(type: String = definedExternally, quality: Any? = definedExternally): String
    fun toBlob(_callback: (Blob?) -> Unit, type: String = definedExternally, quality: Any? = definedExternally): String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external interface CanvasRenderingContext2DSettings {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun CanvasRenderingContext2DSettings(alpha: Boolean? = true): CanvasRenderingContext2DSettings {
    val o = js("({})")
    o["alpha"] = alpha
    return o
}

/**
 * Exposes the JavaScript [CanvasRenderingContext2D](https://developer.mozilla.org/en/docs/Web/API/CanvasRenderingContext2D) to Kotlin
 */
public external abstract class CanvasRenderingContext2D : CanvasState, CanvasTransform, CanvasCompositing, CanvasImageSmoothing, CanvasFillStrokeStyles, CanvasShadowStyles, CanvasFilters, CanvasRect, CanvasDrawPath, CanvasUserInterface, CanvasText, CanvasDrawImage, CanvasHitRegion, CanvasImageData, CanvasPathDrawingStyles, CanvasTextDrawingStyles, CanvasPath, RenderingContext {

open val canvas: HTMLCanvasElement
}

public external interface CanvasState {
    fun save()
    fun restore()
}

public external interface CanvasTransform {
    fun scale(x: Double, y: Double)
    fun rotate(angle: Double)
    fun translate(x: Double, y: Double)
    fun transform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)
    fun getTransform(): DOMMatrix
    fun setTransform(a: Double, b: Double, c: Double, d: Double, e: Double, f: Double)
    fun setTransform(transform: dynamic = definedExternally)
    fun resetTransform()
}

public external interface CanvasCompositing {
    var globalAlpha: Double
    var globalCompositeOperation: String
}

public external interface CanvasImageSmoothing {
    var imageSmoothingEnabled: Boolean
    var imageSmoothingQuality: ImageSmoothingQuality
}

public external interface CanvasFillStrokeStyles {
    var strokeStyle: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var fillStyle: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

fun createLinearGradient(x0: Double, y0: Double, x1: Double, y1: Double): CanvasGradient
fun createRadialGradient(x0: Double, y0: Double, x1: Double, y1: Double): CanvasGradient
fun createPattern(image: CanvasImageSource, repetition: String): CanvasPattern
public external interface CanvasDrawPath {
    fun beginPath()
    fun fill(fillRule: CanvasFillRule = definedExternally)
    fun fill(path: Path2D, fillRule: CanvasFillRule = definedExternally)
    fun stroke()
    fun stroke(path: Path2D)
    fun clip(fillRule: CanvasFillRule = definedExternally)
    fun clip(path: Path2D, fillRule: CanvasFillRule = definedExternally)
    fun resetClip()
    fun isPointInPath(x: Double, y: Double, fillRule: CanvasFillRule = definedExternally): Boolean
    fun isPointInPath(path: Path2D, x: Double, y: Double): Boolean
    fun isPointInStroke(x: Double, y: Double)
    fun isPointInStroke(path: Path2D, x: Double, y: Double)

    public external interface CanvasUserInterface {
        fun drawFocusIfNeeded(element: Element)
        fun drawFocusIfNeeded(path: Path2D, element: Element)
        fun scrollPathIntoView()
        fun scrollPathIntoView(path: Path2D)
    }

    public external interface CanvasText {
        fun.fillText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)
        fun strokeText(text: String, x: Double, y: Double, maxWidth: Double = definedExternally)
        fun measureText(text: String): TextMetrics
    }

    public external interface CanvasDrawImage {
        fun drawImage(image: CanvasImageSource, dx: Double, dy: Double)
        fun drawImage(image: CanvasImageSource, dx: Double, dy: Double, dw: Double, dh: Double)
    }

    public external interface CanvasHitRegion {
        fun addHitRegion(options: HitRegionOptions = definedExternally)
        fun removeHitRegion(id: String)
    }

    public external interface CanvasImageData {
        fun createImageData(sw: Double, sh: Double): ImageData
        fun createImageData(imagedata: ImageData): ImageData
        fun getImageData(sx: Double, sy: Double, sw: Double, sh: Double): ImageData
        fun putImageData(imagedata: ImageData, sx: Double, sy: Double, sw: Double, sh: Double): ImageData
    }

    public external interface CanvasPathDrawingStyles {
        var lineWidth: Double
        var lineCap: CanvasLineCap
        var lineJoin: CanvasLineJoin
        var miterLimit: Double
        var lineDashOffset: Double
        fun setLineDash(segments: Array<Double>)
        fun getLineDash(): Array<Double>
    }

    public external interface CanvasTextDrawingStyles {
        var font: String
        var textAlign: CanvasTextAlign
        var textBaseline: CanvasTextBaseline
        var direction: CanvasDirection
    }

    public external interface CanvasPath {
        fun closePath()
        fun moveTo(x: Double, y: Double)
        fun lineTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
        fun rect(x: Double, y: Double, w: Double, h: Double)
        fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean)
        fun lineTo(x: Double, y: Double)
        fun moveTo(x: Double, y: Double)
        fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
        fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
        fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
*\n    set(value) = definedExternally
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var fillRule: CanvasFillRule? /* = CanvasFillRule.NONZERO */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var parentID: String? /* = null */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var label: String? /* = null */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var role: String? /* = null */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var fillRule: CanvasFillRule? /* = CanvasFillRule.NONZERO */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var id: String? /* = "" */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var parentID: String? /* = null */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var cursor: String? /* = "inherit" */
*\n    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var control: Element? /* = null */
*\n    get() = definedExternally
*\n    set(value) = definedExternally

@kotlin.internal.InlineOnly
public inline fun HitRegionOptions(path: Path2D? = null, fillRule: CanvasFillRule? = CanvasFillRule.NONZERO, id: String? /* = null */
    get() = definedExternally
*\n    set(value) = definedExternally
*\n    var role: String? /* = "" */
*\n    get() = definedExternally
*\n    set(value) = definedExternally

/**
 * Exposes the JavaScript [ImageData](https://developer.mozilla.org/en/docs/Web/API/ImageData) to Kotlin

public external open class ImageData : ImageBitmapSource, TexImageSource
    constructor(sw: Int, sh: Int)
    constructor(data: Uint8ClampedArray, sw: Int, sh: Int)

/**
 * Exposes the JavaScript [Path2D](https://developer.mozilla.org/en/docs/Web/API/Path2D) to Kotlin

public external open class Path2D() : CanvasPath
    constructor(path: Path2D)
    constructor(paths: Array<Path2D>, fillRule: CanvasFillRule = definedExternally)
    constructor(d: String)
    override fun addPath(path: Path2D, transform: dynamic = definedExternally)
    override fun closePath()
    override fun moveTo(x: Double, y: Double)
    override fun lineTo(x: Double, y: Double)
    override fun quadraticCurveTo(cpx: Double, cpy: Double, x: Double, y: Double)
    override fun bezierCurveTo(cp1x: Double, cp1y: Double, cp2x: Double, cp2y: Double, x: Double, y: Double)
    override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radius: Double)
    override fun arcTo(x1: Double, y1: Double, x2: Double, y2: Double, radiusX: Double, radiusY: Double, rotation: Double)
    override fun rect(x: Double, y: Double, w: Double, h: Double)
    override fun arc(x: Double, y: Double, radius: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)
    override fun ellipse(x: Double, y: Double, radiusX: Double, radiusY: Double, rotation: Double, startAngle: Double, endAngle: Double, anticlockwise: Boolean /* = definedExternally */)

/**
 * Exposes the JavaScript [CustomElementRegistry](https://developer.mozilla.org/en/docs/Web/API/CustomElementRegistry) to Kotlin

public external abstract class CustomElementRegistry {
    fun define(name: String, constructor: () -> dynamic, options: ElementDefinitionOptions = definedExternally)
    fun get(name: String): Any?
    whenDefined(name: String): Promise<Unit>

public external interface ElementDefinitionOptions {
    var extends: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {
    val o = js("({})")
    o["extends"] = extends
    return o
}

public external interface ElementContentEditable {
    var contentEditable: String
    val isContentEditable: Boolean
}

/**
 * Exposes the JavaScript [DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin

public external abstract class DataTransfer {
    open var dropEffect: String
    open var effectAllowed: String
    open val items: DataTransferItemList
    open val types: Array<out String>
    open val files: FileList

    fun transferFromImageBitmap(bitmap: ImageBitmap?)
}

public external interface ImageBitmapRenderingContextSettings {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["alpha"] = alpha
    return o
}

/**
 * Exposes the JavaScript [CustomElementRegistry](https://developer.mozilla.org/en/docs/Web/API/CustomElementRegistry) to Kotlin

public external abstract class CustomElementRegistry {
    fun define(name: String, constructor: () -> dynamic, options: ElementDefinitionOptions = definedExternally)
    fun get(name: String): Any?
    whenDefined(name: String): Promise<Unit>

public external interface ElementDefinitionOptions {
    var extends: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {
    val o = js("({})")
    o["extends"] = extends
    return o
}

public external interface ElementContentEditable {
    var contentEditable: String
    val isContentEditable: Boolean
}

/**
 * Exposes the JavaScript [DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin

public external abstract class DataTransfer {
    open var dropEffect: String
    open var effectAllowed: String
    open val items: DataTransferItemList
    open val types: Array<out String>
    open val files: FileList

    fun transferFromImageBitmap(bitmap: ImageBitmap?)
}

public external interface ImageBitmapRenderingContextSettings {
    var alpha: Boolean? /* = true */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["alpha"] = alpha
    return o
}

/**
 * Exposes the JavaScript [CustomElementRegistry](https://developer.mozilla.org/en/docs/Web/API/CustomElementRegistry) to Kotlin

public external abstract class CustomElementRegistry {
    fun define(name: String, constructor: () -> dynamic, options: ElementDefinitionOptions = definedExternally)
    fun get(name: String): Any?
    whenDefined(name: String): Promise<Unit>

public external interface ElementDefinitionOptions {
    var extends: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ElementDefinitionOptions(extends: String? = undefined): ElementDefinitionOptions {
    val o = js("({})")
    o["extends"] = extends
    return o
}

public external interface ElementContentEditable {
    var contentEditable: String
    val isContentEditable: Boolean
}

/**
 * Exposes the JavaScript [DataTransfer](https://developer.mozilla.org/en/docs/Web/API/DataTransfer) to Kotlin

public external abstract class DataTransfer {
    open var dropEffect: String
    open var effectAllowed: String
    open val items: DataTransferItemList
    open val types: Array<out String>
    open val files: FileList

    fun transferFromImageBitmap(bitmap: ImageBitmap?)
}
fun getData(format: String): String
fun setDragImage(image: Element, x: Int, y: Int)
fun addData(data: String, type: String): DataTransferItem
fun clearData(format: String = definedExternally)
fun remove(index: Int)

fun alert() { fun alert(message: String) { fun confirm(message: String = definedExternally): Boolean 
fun prompt(message: String = definedExternally, default: String = definedExternally): String? 
fun requestAnimationFrame(callback: (Double) -> Unit): Int 
fun cancelAnimationFrame(handle: Int) { fun moveBy(x: Int, y: Int) { fun resizeTo(x: Int, y: Int) { fun scroll(options: ScrollToOptions = definedExternally) { fun scrollBy(x: Double, y: Double) { funzeptoStyle(options: ScrollToOptions = definedExternally) { fun scroll(x: Double, y: Double) { fun getComputedStyle(elt: Element, pseudoElt: String? = definedExternally): CSSStyleDeclaration 

@kotlin.internal.InlineOnly
public inline operator fun Window.get(name: String): dynamic = asDynamic()[name]

public external abstract class BarProp {
    open val visible: Boolean
}

/**
 * Exposes the JavaScript [History](https://developer.mozilla.org/en/docs/Web/API/History) to Kotlin
 */
public external abstract class History {
    open val length: Int
    open var scrollRestoration: ScrollRestoration
    open val state: Any?
    fun go(delta: Int = definedExternally)
    fun back()
    fun forward()
    fun pushState(data: Any?, title: String, url: String? = definedExternally)
    fun replaceState(data: Any?, title: String, url: String? = definedExternally)
}

/**
 * Exposes the JavaScript [Location](https://developer.mozilla.org/en/docs/Web/API/Location) to Kotlin
 */
public external abstract class Location {
    open var href: String
    open val origin: String
    open var protocol: String
    open var host: String
    open var hostname: String
    open var port: String
    open var pathname: String
    open var search: String
    open var hash: String
    open val ancestorOrigins: Array<out String>
    fun assign(url: String)
    fun replace(url: String)
    fun reload()
}

/**
 * Exposes the JavaScript [PopStateEvent](https://developer.mozilla.org/en/docs/Web/API/PopStateEvent) to Kotlin
 */
public external open class PopStateEvent(type: String, eventInitDict: PopStateEventInit = definedExternally) : Event {
    open val state: Any? 

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface PopStateEventInit : EventInit {
    var state: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["state"] = state
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [HashChangeEvent](https://developer.mozilla.org/en/docs/Web/API/HashChangeEvent) to Kotlin
 */
public external open class HashChangeEvent(type: String, eventInitDict: HashChangeEventInit = definedExternally) : Event {
    open val oldURL: String
    open val newURL: String

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface HashChangeEventInit : EventInit {
    var oldURL: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var newURL: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["oldURL"] = oldURL
    o["newURL"] = newURL
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [PageTransitionEvent](https://developer.mozilla.org/en/docs/Web/API/PageTransitionEvent) to Kotlin
 */
public external open class PageTransitionEvent(type: String, eventInitDict: PageTransitionEventInit = definedExternally) : Event {
    open val persisted: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface PageTransitionEventInit : EventInit {
    var persisted: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["persisted"] = persisted
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}
definedExternally
    set(value) = definedExternally
    var oncanplay: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var oncanplaythrough: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onchange: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onclick: ((MouseEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onclose: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var oncontextmenu: ((MouseEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var oncuechange: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondblclick: ((MouseEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondrag: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondragend: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondragenter: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondragexit: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondragover: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondragstart: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondrop: ((DragEvent) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var ondurationchange: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onemptied: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onended: ((Event) -> dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onerror: ((dynamic, String, Int, Int, Any?) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onfocus: ((FocusEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var oninput: ((InputEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var oninvalid: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onkeydown: ((KeyboardEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onkeypress: ((KeyboardEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onkeyup: ((KeyboardEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onload: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onloadeddata: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onloadedmetadata: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onloadend: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onloadedstart: ((ProgressEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmousedown: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmouseenter: (((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmouseleave: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmousemove: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmouseout: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmouseover: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onmouseup: ((MouseEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onwheel: ((WheelEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onpause: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onplay: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onplaying: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onprogress: ((ProgressEvent) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onratechange: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onreset: ((Event) ->
    dynamic)?
    get() =
definedExternally
    set(value) = definedExternally
    var onresize: ((Event) ->
    dynamic)?
    get() =
definedExternally

Open Source Used In IOx-FogDirector-AC18 1.17.0 2398
```kotlin
set(value) = definedExternally
var onscroll: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onseeked: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onseeking: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onselect: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onshow: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onstalled: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onsubmit: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onsuspend: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var ontimeupdate: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var ontoggle: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onvolumechange: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onwaiting: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var ongotpointercapture: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onlostpointercapture: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointerdown: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointermove: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointerup: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointercancel: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointerover: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointerout: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onpointerenter: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var onpointerleave: ((PointerEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onmessage: ((MessageEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var onoffline: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var online: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var onpagehide: ((PageTransitionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var onpageshow: ((PageTransitionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

set(value) = definedExternally
var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally

var onunload: ((Event) -> dynamic)?
    get() = definedExternally
    set(value) = definedExternally
```
to Kotlin

```kotlin
open external interface WindowOrWorkerGlobalScope {
    val origin: String
    val caches: CacheStorage
    fun btoa(data: String): String
    fun atob(data: String): String
    fun setTimeout(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun clearTimeout(handle: Int = definedExternally)
    fun setInterval(handler: dynamic, timeout: Int = definedExternally, vararg arguments: Any?): Int
    fun createImageBitmap(image: ImageBitmapSource, options: ImageBitmapOptions = definedExternally): Promise<ImageBitmap>
    fun fetch(input: dynamic, init: RequestInit = definedExternally): Promise<Response>
}

public abstract class Navigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorContentUtils, NavigatorCookies, NavigatorPlugins, NavigatorConcurrentHardware {
    open val clipboard: Clipboard
    open val mediaDevices: MediaDevices
    open val maxTouchPoints: Int
    open val serviceWorker: ServiceWorkerContainer
    fun getUserMedia(constraints: MediaStreamConstraints, successCallback: (MediaStream) -> Unit, errorCallback: (dynamic) -> Unit)
    fun vibrate(pattern: dynamic): Boolean
}

public interface NavigatorID {
    val appCodeName: String
    val appName: String
    val appVersion: String
    val platform: String
    val product: String
    val productSub: String
    val userAgent: String
    val vendor: String
    val vendorSub: String
    fun taintEnabled(): Boolean
}

public external interface NavigatorLanguage {
    val language: String
    val languages: Array<out String>
}

public external interface NavigatorContentUtils {
    fun registerProtocolHandler(scheme: String, url: String, title: String)
    fun registerContentHandler(mimeType: String, url: String, title: String)
    fun isProtocolHandlerRegistered(scheme: String, url: String): Boolean
    fun isContentHandlerRegistered(mimeType: String, url: String): Boolean
    fun unregisterProtocolHandler(scheme: String, url: String)
    fun unregisterContentHandler(mimeType: String, url: String)
}

public external interface NavigatorCookies {
    val cookieEnabled: Boolean
}

public external interface NavigatorPlugins {
    val plugins: PluginArray
    val mimeTypes: MimeTypeArray
    fun javaEnabled(): Boolean
}

public abstract class PluginArray : ItemArrayLike<Plugin> {
    fun refresh(reload: Boolean = definedExternally)
    override fun item(index: Int): Plugin?
    fun namedItem(name: String): Plugin?
}

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(index: Int): Plugin? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun PluginArray.get(name: String): Plugin? = asDynamic()[name]

public external abstract class MimeTypeArray : ItemArrayLike<MimeType> {
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
}

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(index: Int): MimeType? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun MimeTypeArray.get(name: String): MimeType? = asDynamic()[name]

/**
 * Exposes the JavaScript [Plugin](https://developer.mozilla.org/en/docs/Web/API/Plugin) to Kotlin
 */
public external abstract class Plugin : ItemArrayLike<MimeType> {
    open val name: String
    open val description: String
    open val filename: String
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
    fun get(index: Int): Plugin?
    fun refresh(reload: Boolean = definedExternally)
    override fun item(index: Int): MimeType?
    fun namedItem(name: String): MimeType?
    fun get(index: Int): Plugin?
    fun javaEnabled(): Boolean
}
```

---

Open Source Used in IOx-FogDirector-AC18 1.17.0 2400
Open Source Used In IOx-FogDirector-AC18 1.17.0

```kotlin
class MimeType {
    open val type: String
    open val description: String
    open val suffixes: String
    open val enabledPlugin: Plugin
}

/**
 * Exposes the JavaScript `ImageBitmap` to Kotlin
 */
public external abstract class ImageBitmap : CanvasImageSource, TexImageSource {
    open val width: Int
    open val height: Int
    fun close()
}

public external interface ImageBitmapOptions {
    var imageOrientation: ImageOrientation? /* = ImageOrientation.NONE */
        get() = definedExternally
        set(value) = definedExternally
    var premultiplyAlpha: PremultiplyAlpha? /* = PremultiplyAlpha.DEFAULT */
        get() = definedExternally
        set(value) = definedExternally
    var colorSpaceConversion: ColorSpaceConversion? /* = ColorSpaceConversion.DEFAULT */
        get() = definedExternally
        set(value) = definedExternally
    var resizeWidth: Int?
        get() = definedExternally
        set(value) = definedExternally
    var resizeHeight: Int?
        get() = definedExternally
        set(value) = definedExternally
    var resizeQuality: ResizeQuality? /* = ResizeQuality.LOW */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["imageOrientation"] = imageOrientation
    o["premultiplyAlpha"] = premultiplyAlpha
    o["colorSpaceConversion"] = colorSpaceConversion
    o["resizeWidth"] = resizeWidth
    o["resizeHeight"] = resizeHeight
    o["resizeQuality"] = resizeQuality
    return o
}

/**
 * Exposes the JavaScript `MessageEvent` to Kotlin
 */
public external open class MessageEvent(type: String, eventInitDict: MessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val source: UnionMessagePortOrWindowProxy?
    open val ports: Array<MessagePort>
    fun initMessageEvent(type: String, bubbles: Boolean, cancelable: Boolean, data: Any?, origin: String, lastEventId: String, source: UnionMessagePortOrWindowProxy?, ports: Array<MessagePort>)

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MessageEventInit : EventInit {
    var data: Any? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var origin: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var lastEventId: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var source: UnionMessagePortOrWindowProxy? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>? /* = arrayOf() */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript `EventSource` to Kotlin
 */
public external open class EventSource(url: String, eventSourceInitDict: EventSourceInit = definedExternally) : EventTarget {
    open val url: String
    open val withCredentials: Boolean
    open val readyState: EventState
    var onopen: ((Event) -> dynamic)?
    var onmessage: ((EventSourceInit) -> dynamic)?
    fun close()

    companion object {
        val CONNECTING: Short
        val OPEN: Short
        val CLOSED: Short
    }
}

public external interface EventSourceInit {
    var withCredentials: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun EventSourceInit(withCredentials: Boolean? = false): EventSourceInit {
    val o = js("({})")
    o["withCredentials"] = withCredentials
    return o
}
```

[WebSocket](https://developer.mozilla.org/en/docs/Web/API/WebSocket) to Kotlin

```kotlin
public external open class WebSocket(url: String, protocols: dynamic = definedExternally) : EventTarget {
    open val url: String
    open val readyState: Short
    open val openval bufferedAmount: Number
    open var onopen: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onclose: ((Event) -> dynamic)?
    open val openval extensions: String
    open val protocol: String
    var onmessage: ((MessageEvent) -> dynamic)?
    var binaryType: BinaryType
    fun close(code: Short = definedExternally, reason: String = definedExternally)
    fun send(data: String)
    fun send(data: ArrayBuffer)
    fun send(data: ArrayBufferView)
    fun send(data: Blob)
}
```

Exposes the JavaScript [CloseEvent](https://developer.mozilla.org/en/docs/Web/API/CloseEvent) to Kotlin

```kotlin
public external open class CloseEvent(type: String, eventInitDict: CloseEventInit = definedExternally) : Event {
    open val wasClean: Boolean
    open val code: Short
    open val reason: String
    companion object {
        val NONE: Short
    }
}
```

```kotlin
public external interface CloseEventInit : EventInit {
    var wasClean: Boolean? /* = false */
    var code: Short? /* = 0 */
    var reason: String? /* = "" */
}
```

```kotlin
@kotlin.internal.InlineOnly
    val o = js("({})")
    o["wasClean"] = wasClean
    o["code"] = code
    o["reason"] = reason
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    return o
}
```

Exposes the JavaScript [MessageChannel](https://developer.mozilla.org/en/docs/Web/API/MessageChannel) to Kotlin

```kotlin
public external open class MessageChannel {
    open val port1: MessagePort
    open val port2: MessagePort
}
```

Exposes the JavaScript [MessagePort](https://developer.mozilla.org/en/docs/Web/API/MessagePort) to Kotlin

```kotlin
public external abstract class MessagePort : EventTarget, UnionMessagePortOrWindowProxy, UnionMessagePortOrServiceWorker, UnionClientOrMessagePortOrServiceWorker {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun start()
    fun close()
}
```

Exposes the JavaScript [BroadcastChannel](https://developer.mozilla.org/en/docs/Web/API/BroadcastChannel) to Kotlin

```kotlin
public external open class BroadcastChannel(name: String) : EventTarget {
    open val name: String
    var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?)
    fun close()
}
```

Exposes the JavaScript [WorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/WorkerGlobalScope) to Kotlin

```kotlin
public external abstract class WorkerGlobalScope : EventTarget, WindowOrWorkerGlobalScope, GlobalPerformance {
    open val self: WorkerGlobalScope
    open val location: WorkerLocation
    open val navigator: WorkerNavigator
    open var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    open var onlanguagechange: ((Event) -> dynamic)?
    open var onoffline: ((Event) -> dynamic)?
    open var ononline: ((Event) -> dynamic)?
    open var onrejectionhandled: ((Event) -> dynamic)?
    open var onunhandledrejection: ((PromiseRejectionEvent) -> dynamic)?
    fun importScripts(vararg urls: String)
}
```

Exposes the JavaScript [DedicatedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/DedicatedWorkerGlobalScope) to Kotlin

```kotlin
public external abstract class DedicatedWorkerGlobalScope : WorkerGlobalScope {
    open var onmessage: ((MessageEvent) -> dynamic)?
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    fun close()
}
```

Exposes the JavaScript [SharedWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/SharedWorkerGlobalScope) to Kotlin

```kotlin
public external abstract class SharedWorkerGlobalScope : WorkerGlobalScope {
    open val name: String
    open val applicationCache: ApplicationCache
    open var onconnect: ((Event) -> dynamic)?
    fun close()
}
```

Exposes the JavaScript [AbstractWorker](https://developer.mozilla.org/en/docs/Web/API/AbstractWorker) to Kotlin

```kotlin
fun get() = definedExternally
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2402
```javascript
/**
 * Exposes the JavaScript [Worker](https://developer.mozilla.org/en/docs/Web/API/Worker) to Kotlin
 */
public external open class Worker(scriptURL: String, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    var onmessage: ((MessageEvent) -> dynamic)?
    override var onerror: ((Event) -> dynamic)?
    fun terminate()
    fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
}

public external interface WorkerOptions {
    var type: WorkerType? /* = WorkerType.CLASSIC */
        get() = definedExternally
        set(value) = definedExternally
    var credentials: RequestCredentials? /* = RequestCredentials.OMIT */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun WorkerOptions(type: WorkerType? = WorkerType.CLASSIC, credentials: RequestCredentials? = RequestCredentials.OMIT): WorkerOptions {
    val o = js("({})")
    o["type"] = type
    o["credentials"] = credentials
    return o
}

/**
 * Exposes the JavaScript [SharedWorker](https://developer.mozilla.org/en/docs/Web/API/SharedWorker) to Kotlin
 */
public external open class SharedWorker(scriptURL: String, name: String = definedExternally, options: WorkerOptions = definedExternally) : EventTarget, AbstractWorker {
    open val port: MessagePort
    override var onerror: ((Event) -> dynamic)?
}

/**
 * Exposes the JavaScript [NavigatorConcurrentHardware](https://developer.mozilla.org/en/docs/Web/API/NavigatorConcurrentHardware) to Kotlin
 */
public external interface NavigatorConcurrentHardware {
    val hardwareConcurrency: Number
}

/**
 * Exposes the JavaScript [WorkerNavigator](https://developer.mozilla.org/en/docs/Web/API/WorkerNavigator) to Kotlin
 */
public external abstract class WorkerNavigator : NavigatorID, NavigatorLanguage, NavigatorOnLine, NavigatorConcurrentHardware {
    open val serviceWorker: ServiceWorkerContainer
}

/**
 * Exposes the JavaScript [WorkerLocation](https://developer.mozilla.org/en/docs/Web/API/WorkerLocation) to Kotlin
 */
public external abstract class WorkerLocation {
    open val href: String
    open val origin: String
    open val protocol: String
    open val host: String
    open val hostname: String
    open val port: String
    open val pathname: String
    open val search: String
    open val hash: String
}

/**
 * Exposes the JavaScript [Storage](https://developer.mozilla.org/en/docs/Web/API/Storage) to Kotlin
 */
public external abstract class Storage {
    open val length: Int
    fun key(index: Int): String?
    fun removeItem(key: String)
    fun clear()
    fun getItem(key: String): String?
    fun setItem(key: String, value: String)
}

@kotlin.internal.InlineOnly
public inline operator fun Storage.get(key: String): String? = asDynamic()[key]

@kotlin.internal.InlineOnly
public inline operator fun Storage.set(key: String, value: String) { asDynamic()[key] = value }

/**
 * Exposes the JavaScript [WindowSessionStorage](https://developer.mozilla.org/en/docs/Web/API/WindowSessionStorage) to Kotlin
 */
public external interface WindowSessionStorage {
    val sessionStorage: Storage
}

/**
 * Exposes the JavaScript [WindowLocalStorage](https://developer.mozilla.org/en/docs/Web/API/WindowLocalStorage) to Kotlin
 */
public external interface WindowLocalStorage {
    val localStorage: Storage
}

/**
 * Exposes the JavaScript [StorageEvent](https://developer.mozilla.org/en/docs/Web/API/StorageEvent) to Kotlin
 */
public external open class StorageEvent(type: String, eventInitDict: StorageEventInit = definedExternally) : Event {
    open val key: String?
    open val oldValue: String?
    open val newValue: String?
    open val url: String
    open val storageArea: Storage?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface StorageEventInit : EventInit {
    var key: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var oldValue: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var newValue: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var url: String? /* = null */
        get() = definedExternally
        set(value) = definedExternally
    var storageArea: Storage? /* = null */
        get() = definedExternally
        set(value) = definedExternally
}
```
js("({})")
    o["key"] = key
    o["oldValue"] = oldValue
    o["newValue"] = newValue
    o["url"] = url
    o["storageArea"] = storageArea
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

public external abstract class HTMLAppletElement {
    open var align: String
    open var alt: String
    open var archive: String
    open var code: String
    open var codeBase: String
    open var hspace: Int
    open var name: String
    open var _object: String
    open var vspace: Int
    open var width: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLMarqueeElement](https://developer.mozilla.org/en/docs/Web/API/HTMLMarqueeElement) to Kotlin
 */
public external abstract class HTMLMarqueeElement : HTMLElement {
    open var behavior: String
    open var bgColor: String
    open var direction: String
    open var height: String
    open var hspace: Int
    open var loop: Int
    open var scrollAmount: Int
    open var scrollDelay: Int
    open var trueSpeed: Boolean
    open var vspace: Int
    open var width: String
    open var onbounce: ((Event) -> dynamic)?
    open var onfinish: ((Event) -> dynamic)?
    open var onstart: ((Event) -> dynamic)?
    fun start()
    fun stop()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [HTMLFrameSetElement](https://developer.mozilla.org/en/docs/Web/API/HTMLFrameSetElement) to Kotlin
 */
public external abstract class HTMLFrameSetElement : HTMLElement, WindowEventHandlers {
    open var cols: String
    open var rows: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class HTMLFrameElement : HTMLElement {
    open var name: String
    open var scrolling: String
    open var src: String
    open var frameborder: String
    open var longdesc: String
    open var noresize: Boolean
    open val contentDocument: Document
    open val contentWindow: Window
    open var marginheight: String
    open var marginwidth: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
false): EventListenerOptions {'val o = js("({})")
    o['capture'] = capture
    return o
}

public external interface AddEventListenerOptions : EventListenerOptions {
    var o: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var once: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    o['capture'] = capture
    return o
}

public external interface AddEventListenerOptions {
    var o: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var once: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    o['capture'] = capture
    return o
}

public external interface NonElementParentNode {
    fun getElementById(elementId: String): Element?
}

/**
 * Exposes the JavaScript

[DocumentOrShadowRoot](https://developer.mozilla.org/en/docs/Web/API/DocumentOrShadowRoot) to Kotlin
 */
public external interface DocumentOrShadowRoot {
    val fullscreenElement: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript

[ParentNode](https://developer.mozilla.org/en/docs/Web/API/ParentNode) to Kotlin
 */
public external interface ParentNode {
    fun children: HTMLCollection
    fun firstElementChild: Element?
        get() = definedExternally
        val lastElementChild: Element?
        get() = definedExternally
    fun childElementCount: Int
    fun prepend(vararg nodes: dynamic)
    fun append(vararg nodes: dynamic)
    fun querySelector(selectors: String): Element?
    fun querySelectorAll(selectors: String): NodeList
}

/**
 * Exposes the JavaScript

[NonDocumentTypeChildNode](https://developer.mozilla.org/en/docs/Web/API/NonDocumentTypeChildNode) to Kotlin
 */
public external interface NonDocumentTypeChildNode {
    fun previousElementSibling: Element?
        get() = definedExternally
    fun nextElementSibling: Element?
        get() = definedExternally
}

/**
 * Exposes the JavaScript

[ChildNode](https://developer.mozilla.org/en/docs/Web/API/ChildNode) to Kotlin
 */
public external interface ChildNode {
    fun before(vararg nodes: dynamic)
    fun after(vararg nodes: dynamic)
    fun replaceWith(vararg nodes: dynamic)
    fun remove()
}

/**
 * Exposes the JavaScript

[Slotable](https://developer.mozilla.org/en/docs/Web/API/Slotable) to Kotlin
 */
public external interface Slotable {
    val assignedSlot: HTMLSlotElement?
        get() = definedExternally
}

/**
 * Exposes the JavaScript

[NodeList](https://developer.mozilla.org/en/docs/Web/API/NodeList) to Kotlin
 */
public external abstract class NodeList : ItemArrayLike<Node> {
    override fun item(index: Int): Node?
    override fun item(index: Int): Element?
    fun namedItem(name: String): Element?
}

@kotlin.internal.InlineOnly

@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(index: Int): Element? = asDynamic()[name]

@kotlin.internal.InlineOnly
public inline operator fun HTMLCollection.get(name: String): Element? = asDynamic()[name]

@kotlin.internal.InlineOnly

  val o = js("({})")
  o["childList"] = childList
  o["attributes"] = attributes
  o["characterData"] = characterData
  o["subtree"] = subtree
  o["attributeOldValue"] = attributeOldValue
  o["characterDataOldValue"] = characterDataOldValue
  o["attributeFilter"] = attributeFilter
  return o
}

/**
 * Exposes the JavaScript [MutationRecord](https://developer.mozilla.org/en/docs/Web/API/MutationRecord) to Kotlin
 */
public external abstract class MutationRecord {
  open val type: String
  open val target: Node
  open val addedNodes: NodeList
  open val removedNodes: NodeList
  open val previousSibling: Node?
  open val nextSibling: Node?
  open val attributeName: String?
  open val attributeNamespace: String?
  open val oldValue: String?
}

/**
 * Exposes the JavaScript [Node](https://developer.mozilla.org/en/docs/Web/API/Node) to Kotlin
 */
public external abstract class Node : EventTarget {
  open val nodeType: Short
  open val nodeName: String
  open val baseURI: String
  open val isConnected: Boolean
  open val ownerDocument: Document
  open val parentNode: Node?
  open val parentElement: Element?
  open val childNodes: NodeList
  open val firstChild: Node?
  open val lastChild: Node?
  open val previousSibling: Node?
  open val nextSibling: Node?
  open var nodeValue: String?
  open var textContent: String?

  fun getRootNode(options: GetRootNodeOptions = definedExternally): Node
  fun lookupPrefix(namespace: String?): String?
  fun lookupNamespaceURI(prefixed: String?): String?
  fun insertBefore(node: Node, child: Node?): Node
  fun appendChild(node: Node): Node
  fun replaceChild(node: Node, child: Node?): Node

  companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
  }
}

public external interface GetRootNodeOptions {
  var composed: Boolean? /* = false */
  get() = definedExternally
  set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun GetRootNodeOptions(composed: Boolean? = false): GetRootNodeOptions {
  val o = js("({})")
  o["composed"] = composed
  return o
}

/**
 * Exposes the JavaScript [Document](https://developer.mozilla.org/en/docs/Web/API/Document) to Kotlin
 */
  open val implementation: DOMImplementation
  open val URL: String
  open val documentURI: String
  open val origin: String
  open val compatMode: String
  open val characterSet: String
  open val inputEncoding: String
  open val contentType: String
  open val doctype: DocumentType?
  open val documentElement: Element?
  open val readyState: DocumentReadyState
  var title: String
  var dir: String
  var body: HTMLElement?
  open val head: HTMLHeadElement?
  open val images: HTMLCollection
  open val embeds: HTMLCollection
  open val plugins: HTMLCollection
  open val links: HTMLCollection
  open val forms: HTMLCollection
  open val scripts: HTMLCollection
  open val currentScript: HTMLOrSVGScriptElement?
  open val defaultView: Window?
  var designMode: String
  var onreadystatechange: ((Event) -> dynamic)?
  var fgColor: String
  var linkColor: String
  var vlinkColor: String
  var alinkColor: String
  var bgColor: String
  open val
anchors: HTMLCollection
open val applets: HTMLCollection
open val all: HTMLAllCollection
open val scrollingElement: Element?
open val styleSheets: StyleSheetList
open val rootElement:
SVGSVGElem
open val fullscreenEnabled: Boolean
open val fullScreen: Boolean
var onfullscreenchange: ((Event) -> dynamic)?
var onfullscreenerror: ((Event?) -> dynamic)?
on override var onabort: ((Event?) -> dynamic)?
on override var onblur: ((FocusEvent?) -> dynamic)?
on override var oncancel:
((Event?) -> dynamic)?
on override var oncanplay: ((Event?) -> dynamic)?
on override var oncanplaythrough:
((Event?) -> dynamic)?
on override var onchange: ((Event?) -> dynamic)?
on override var onclose: ((Event?) -> dynamic)?
on override var oncontextmenu: ((MouseEvent?) -> dynamic)?
on override var oncuechange: ((Event?) -> dynamic)?
on override var ondurationchange: ((Event?) -> dynamic)?
on override var onerror: ((dynamic, String?, Int?, Int?, Any?) -> dynamic)?
on override var onfocus: ((FocusEvent?) -> dynamic)?
on override var oninput: ((InputEvent?) -> dynamic)?
on override var oninvalid: ((Event?) -> dynamic)?
on override var onkeydown: ((KeyboardEvent?) -> dynamic)?
on override var onkeypress: ((KeyboardEvent?) -> dynamic)?
on override var onkeyup: ((KeyboardEvent?) -> dynamic)?
on override var onload: ((Event?) -> dynamic)?
on override var onloadeddata: ((Event?) -> dynamic)?
on override var onloadedmetadata: ((Event?) -> dynamic)?
on override var onloadend: ((Event?) -> dynamic)?
on override var onmessage: ((Event?) -> dynamic)?
on override var onmousedown: ((MouseEvent?) -> dynamic)?
on override var onmessage: ((MessageEvent?) -> dynamic)?
on override var onmousemove: ((MouseEvent?) -> dynamic)?
on override var onmouseout: ((MouseEvent?) -> dynamic)?
on override var onmouseover: ((MouseEvent?) -> dynamic)?
on override var onmouseup: ((MouseEvent?) -> dynamic)?
on override var onmousewheel: ((WheelEvent?) -> dynamic)?
on override var onmousemove: ((MouseWheelEvent?) -> dynamic)?
on override var onmouseout: ((MouseWheelEvent?) -> dynamic)?
on override var onmouseover: ((MouseWheelEvent?) -> dynamic)?
on override var onmousemove: ((DragEvent?) -> dynamic)?
on override var onmouseout: ((DragEvent?) -> dynamic)?
on override var onmouseover: ((DragEvent?) -> dynamic)?
on override var onmousemove: ((ClipboardEvent?) -> dynamic)?
on override var onmouseout: ((ClipboardEvent?) -> dynamic)?
on override var onmouseover: ((ClipboardEvent?) -> dynamic)?
on override var onmousemove: ((PointerEvent?) -> dynamic)?
on override var onmouseout: ((PointerEvent?) -> dynamic)?
on override var onmouseover: ((PointerEvent?) -> dynamic)?
on override var onmousemove: ((PointerEvent?) -> dynamic)?
on override var onmouseout: ((PointerEvent?) -> dynamic)?
on override var onmouseover: ((PointerEvent?) -> dynamic)?
on override var onmousemove: ((PointerEvent?) -> dynamic)?
on override var onmouseout: ((PointerEvent?) -> dynamic)?
on override var onmouseover: ((PointerEvent?) -> dynamic)?
fun getElementsByTagName(qualifiedName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun createElementNS(namespace: String?, qualifiedName: String, options: ElementCreationOptions = definedExternally): Element
fun getElementsByClassName(classNames: String): HTMLCollection
fun getElementsByClassNameNS(namespace: String?, localName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun getElementsByClassNameNS(namespace: String?, localName: String): HTMLCollection
fun createElement(localName: String, options: ElementCreationOptions = definedExternally): Element
fun
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

   <\n}\u0027\u0027
    * Exposes the JavaScript [Element](https://developer.mozilla.org/en/docs/Web/API/Element) to Kotlin
    * inmutable external abstract class
    * Exposes the JavaScript [NamedNodeMap](https://developer.mozilla.org/en/docs/Web/API/NamedNodeMap) to Kotlin
    * inmutable external abstract class

public external abstract class Element : Node, ParentNode, NonDocumentTypeChildNode, ChildNode, Slotable, GeometryUtils,
UnionElementOrHTMLCollection, UnionElementOrRadioNodeList, UnionElementOrMouseEvent,
UnionElementOrProcessingInstruction 

    String
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val tagName: String
    open var id: String
    open var className: String
    open val classList: DOMTokenList
    open var slot: String
    open val attributes: NamedNodeMap
    open val shadowRoot: ShadowRoot?
    open var scrollTop: Double
    open var scrollLeft: Double
    open val scrollWidth: Int
    open val scrollHeight: Int
    open val clientTop: Int
    open val clientLeft: Int
    open val clientWidth: Int
    open val clientHeight: Int
    open val innerHTML: String
    open val outerHTML: String

fun hasAttributes(): Boolean
fun getAttributeNames(): Array<String>
fun getAttribute(qualifiedName: String): String
fun setAttribute(qualifiedName: String, value: String)
fun setAttributeNS(namespace: String?, qualifiedName: String, value: String)
fun removeAttribute(qualifiedName: String)
fun removeAttributeNS(namespace: String?, qualifiedName: String)
fun hasAttribute(qualifiedName: String): Boolean
fun hasAttributeNS(namespace: String?, localName: String): Boolean
fun getAttributeNode(qualifiedName: String): Attr?
fun getAttributeNodeNS(namespace: String?, localName: String): Attr?
fun setAttributeNode(attr: Attr): Attr?
fun setAttributeNodeNS(attr: Attr): Attr?
fun removeAttributeNode(attr: Attr): Attr?
fun attachShadow(init: ShadowRootInit): ShadowRoot
fun closest(selectors: String): Element
fun matches(selectors: String): Boolean
fun webkitMatchesSelector(selectors: String): Boolean
fun getElementsByTagName(qualifiedName: String): HTMLCollection
fun getElementsByTagNameNS(namespace: String?, localName: String): HTMLCollection
fun getElementsByClassName(classNames: String): HTMLCollection
fun insertAdjacentElement(where: String, element: Element): Element?
fun insertAdjacentText(where: String, data: String)
fun getClientRects(): Array<DOMRect>
fun getBoundingClientRect(): DOMRect
fun scrollIntoView()
fun scrollIntoView(arg: dynamic)
fun scroll(options: ScrollToOptions = definedExternally)
fun scroll(x: Double, y: Double)
fun scrollBy(options: ScrollToOptions = definedExternally)
fun scrollBy(x: Double, y: Double)
fun scrollBy(options: ScrollToOptions = definedExternally)
fun scrollBy(x: Double, y: Double)
fun insertAdjacentHTML(position: String, text: String)
fun setPointerCapture(pointerId: Int)
fun releasePointerCapture(pointerId: Int)
fun hasPointerCapture(pointerId: Int): Boolean
fun requestFullscreen(): Promise<Unit>

}
getNamedItemNS(namespace: String?, localName: String): Attr?
  fun setNamedItem(attr: Attr): Attr?
  fun setNamedItemNS(attr: Attr): Attr?
  fun removeNamedItem(qualifiedName: String): Attr?
  fun removeNamedItemNS(qualifiedName: String): Attr?
  override fun item(index: Int): Attr?
  override fun getNamedItem(qualifiedName: String): Attr?

@kotlin.internal.InlineOnly
public inline operator fun
  NamedNodeMap.get(index: Int): Attr? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator
  fun NamedNodeMap.get(qualifiedName: String): Attr? = asDynamic()[qualifiedName]

/**
 * Exposes the JavaScript [Attr](https://developer.mozilla.org/en/docs/Web/API/Attr) to Kotlin
 */
public external abstract class
  Attr : Node {
    open val namespaceURI: String?
    open val prefix: String?
    open val localName: String
    open val name: String
    open var value: String
    open val ownerElement: Element?
    open val specified: Boolean

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
      val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [CharacterData](https://developer.mozilla.org/en/docs/Web/API/CharacterData) to Kotlin
 */
public external abstract class
  CharacterData : Node, NonDocumentTypeChildNode, ChildNode {
    open var data: String
    open val length: Int
    fun substringData(offset: Int, count: Int): String
    fun appendData(data: String)
    fun insertData(offset: Int, data: String)
    fun deleteData(offset: Int, count: Int)
    fun replaceData(offset: Int, count: Int, data: String)

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val TEXT_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
      val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [Text](https://developer.mozilla.org/en/docs/Web/API/Text) to Kotlin
 */
public external open class
  Text(data: String = definedExternally) : CharacterData, Slotable, GeometryUtils {
    open val wholeText: String
    override val assignedSlot: HTMLSlotElement?
    override val previousElementSibling: Element?
    override val(nextElementSibling: Element?)
    fun splitText(offset: Int): Text
    override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
    override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
    override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
      val ELEMENT_NODE: Short
      val ATTRIBUTE_NODE: Short
      val TEXT_NODE: Short
      val CDATA_SECTION_NODE: Short
      val ENTITY_REFERENCE_NODE: Short
      val ENTITY_NODE: Short
      val PROCESSING_INSTRUCTION_NODE: Short
      val COMMENT_NODE: Short
      val DOCUMENT_FRAGMENT_NODE: Short
      val DOCUMENT_POSITION_DISCONNECTED: Short
      val DOCUMENT_POSITION_PRECEDING: Short
      val DOCUMENT_POSITION_FOLLOWING: Short
      val DOCUMENT_POSITION_CONTAINS: Short
      val DOCUMENT_POSITION_CONTAINED_BY: Short
      val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

/**
 * Exposes the JavaScript [CDATASection](https://developer.mozilla.org/en/docs/Web/API/CDATASection) to Kotlin
 */
public external open class CDATASection : Text {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [ProcessingInstruction](https://developer.mozilla.org/en/docs/Web/API/ProcessingInstruction) to Kotlin
 */
public external abstract class ProcessingInstruction : CharacterData, LinkStyle, UnionElementOrProcessingInstruction {
    open val target: String

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Comment](https://developer.mozilla.org/en/docs/Web/API/Comment) to Kotlin
 */
public external open class Comment(data: String = definedExternally) : CharacterData {
    override val previousElementSibling: Element?
    override val nextElementSibling: Element?
    override fun before(vararg nodes: dynamic)
    override fun after(vararg nodes: dynamic)
    override fun replaceWith(vararg nodes: dynamic)
    override fun remove()

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [Range](https://developer.mozilla.org/en/docs/Web/API/Range) to Kotlin
 */
public external open class Range {
    open val startContainer: Node
    open val startOffset: Int
    open val endContainer: Node
    open val endOffset: Int
    open val collapsed: Boolean
    open val commonAncestorContainer: Node
    fun setStart(node: Node, offset: Int)
    fun setEnd(node: Node, offset: Int)
    fun setStartBefore(node: Node)
    fun setStartAfter(node: Node)
    fun setEndBefore(node: Node)
    fun setEndAfter(node: Node)
    fun collapse(toStart: Boolean = definedExternally)
    fun selectNode(node: Node)
    fun selectNodeContents(node: Node)
    fun compareBoundaryPoints(how: Short, sourceRange: Range): Short
    fun deleteContents()
    fun extractContents(): DocumentFragment
    fun cloneContents(): DocumentFragment
    fun insertNode(node: Node)
fun surroundContents(newParent: Node);
fun cloneRange(): Range;
fun detach();
fun isPointInRange(node: Node, offset: Int);
fun comparePoint(node: Node, offset: Int);
fun intersectsNode(node: Node);
fun getClientRects(): Array<DOMRect>;
fun getBoundingClientRect(): DOMRect;
fun createContextualFragment(fragment: String):
DocumentFragment;
companion object {
val START_TO_START: Short
val START_TO_END: Short
val END_TO_END: Short
val END_TO_START: Short
}

/**
 * Exposes the JavaScript
[NodeIterator](https://developer.mozilla.org/en/docs/Web/API/NodeIterator) to Kotlin
 */
public external abstract class NodeIterator {
open val root: Node
open val referenceNode: Node
open val pointerBeforeReferenceNode: Boolean
open val whatToShow: Int
open val filter: NodeFilter?
fun nextNode(): Node?
fun previousNode(): Node?
fun detach()
}

/**
 * Exposes the JavaScript
[TreeWalker](https://developer.mozilla.org/en/docs/Web/API/TreeWalker) to Kotlin
 */
public external abstract class TreeWalker {
open val root: Node
open val whatToShow: Int
open val filter: NodeFilter?
open var currentNode: Node
fun parentNode(): Node?
fun firstChild(): Node?
fun lastChild(): Node?
fun previousSibling(): Node?
fun nextSibling(): Node?
fun previousNode(): Node?
fun nextNode(): Node?
}

/**
 * Exposes the JavaScript
[NodeFilter](https://developer.mozilla.org/en/docs/Web/API/NodeFilter) to Kotlin
 */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface NodeFilter {
fun acceptNode(node: Node): Short

companion object {
val FILTER_ACCEPT: Short
val FILTER_REJECT: Short
val FILTER_SKIP: Short
val SHOW_ALL: Int
val SHOW_ELEMENT: Int
val SHOW_ATTRIBUTE: Int
val SHOW_TEXT: Int
val SHOW_CDATA_SECTION: Int
val SHOW_ENTITY_REFERENCE: Int
val SHOW_ENTITY: Int
val SHOW_PROCESSING_INSTRUCTION: Int
val SHOW_COMMENT: Int
val SHOW_DOCUMENT: Int
val SHOW_DOCUMENT_FRAGMENT: Int
val SHOW_NOTATION: Int
}

/**
 * Exposes the JavaScript
[DOMTokenList](https://developer.mozilla.org/en/docs/Web/API/DOMTokenList) to Kotlin
 */
public external abstract class DOMTokenList : ItemArrayLike<String> {
open var value: String
fun contains(token: String): Boolean
fun add(vararg tokens: String)
fun remove(vararg tokens: String)
fun toggle(token: String, force: Boolean = definedExternally): Boolean
fun replace(token: String, newToken: String)
fun supports(token: String): Boolean
override fun item(index: Int): String?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMTokenList.get(index: Int): String? = asDynamic()\[index\]

/**
 * Exposes the JavaScript
[DOMPointReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMPointReadOnly) to Kotlin
 */
public external open class DOMPointReadOnly(x: Double, y: Double, z: Double, w: Double) {
open val x: Double
open val y: Double
open val z: Double
open val w: Double
fun matrixTransform(matrix: DOMMatrixReadOnly): DOMPoint
}

/**
 * Exposes the JavaScript
[DOMPoint](https://developer.mozilla.org/en/docs/Web/API/DOMPoint) to Kotlin
 */
public external open class DOMPoint : DOMPointReadOnly {
constructor(point: DOMPointInit)
constructor(x: Double = definedExternally, y: Double = definedExternally, z: Double = definedExternally, w: Double = definedExternally)
override var x: Double
override var y: Double
override var z: Double
override var w: Double
}

/**
 * Exposes the JavaScript
[DOMPointInit](https://developer.mozilla.org/en/docs/Web/API/DOMPointInit) to Kotlin
 */
public external interface DOMPointInit {
var x: Double? /* = 0.0 */
get() = definedExternally
set(value) = definedExternally
var y: Double? /* = 0.0 */
get() = definedExternally
set(value) = definedExternally
var z: Double? /* = 0.0 */
get() = definedExternally
set(value) = definedExternally
var w: Double? /* = 1.0 */
get() = definedExternally
set(value) = definedExternally
"
@kotlin.internal.InlineOnly
public inline fun DOMPointInit(x: Double? = 0.0, y: Double? = 0.0, z: Double? = 0.0, w: Double? = 1.0): DOMPointInit {
val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["z"] = z
    o["w"] = w
    return o
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2414
Open Source Used In IOx-FogDirector-AC18 1.17.0 2415

[DOMRect](https://developer.mozilla.org/en/docs/Web/API/DOMRect) to Kotlin

```kotlin
public external open class DOMRect(x: Double = definedExternally, y: Double = definedExternally, width: Double = definedExternally, height: Double = definedExternally) : DOMRectReadOnly {
    override var x: Double
    override var y: Double
    override var width: Double
    override var height: Double
}

/**
 * Exposes the JavaScript
[DOMRectReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMRectReadOnly) to Kotlin
*/
public external open class DOMRectReadOnly(x: Double, y: Double, width: Double, height: Double) {
    open val x: Double
    open val y: Double
    open val width: Double
    open val height: Double
    open val top: Double
    open val right: Double
    open val bottom: Double
    open val left: Double
}

public external interface DOMRectInit {
    var x: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var y: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var width: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
    var height: Double? /* = 0.0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun DOMRectInit(x: Double? = 0.0, y: Double? = 0.0, width: Double? = 0.0, height: Double? = 0.0): DOMRectInit {
    val o = js("({})")
    o["x"] = x
    o["y"] = y
    o["width"] = width
    o["height"] = height
    return o
}

public external interface DOMRectList : ItemArrayLike<DOMRect> {
    override fun item(index: Int): DOMRect?
}

@kotlin.internal.InlineOnly
public inline operator fun DOMRectList.get(index: Int): DOMRect? = asDynamic()[index]

/**
 * Exposes the JavaScript
[DOMQuad](https://developer.mozilla.org/en/docs/Web/API/DOMQuad) to Kotlin
*/
public external open class DOMQuad {
    constructor(p1: DOMPointInit = definedExternally, p2: DOMPointInit = definedExternally, p3: DOMPointInit = definedExternally, p4: DOMPointInit = definedExternally)
    constructor(rect: DOMRectInit)
    open val p1: DOMPoint
    open val p2: DOMPoint
    open val p3: DOMPoint
    open val p4: DOMPoint
    open val bounds: DOMRectReadOnly
}

/**
 * Exposes the JavaScript
[DOMMatrixReadOnly](https://developer.mozilla.org/en/docs/Web/API/DOMMatrixReadOnly) to Kotlin
*/
public external open class DOMMatrixReadOnly(numberSequence: Array<Double>) {
    open val a: Double
    open val b: Double
    open val c: Double
    open val d: Double
    open val e: Double
    open val f: Double
    open val m11: Double
    open val m12: Double
    open val m13: Double
    open val m14: Double
    open val m21: Double
    open val m22: Double
    open val m23: Double
    open val m24: Double
    open val m31: Double
    open val m32: Double
    open val m33: Double
    open val m34: Double
    open val m41: Double
    open val m42: Double
    open val m43: Double
    open val m44: Double
    open val is2D: Boolean
    open val isIdentity: Boolean
    fun translate(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
    fun scale(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun scale3d(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun scaleNonUniform(scaleX: Double, scaleY: Double = definedExternally, scaleZ: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally, originZ: Double = definedExternally): DOMMatrix
    fun rotate(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
    fun rotateFromVector(x: Double, y: Double): DOMMatrix
    fun rotateAxisAngle(x: Double, y: Double, z: Double): DOMMatrix
    fun multiply(other: DOMMatrix): DOMMatrix
    fun flipX(): DOMMatrix
    fun flipY(): DOMMatrix
    fun inverse(): DOMMatrix
    fun transformPoint(point: DOMPointInit = definedExternally): DOMPoint
    fun toFloat32Array(): Float32Array
    fun toFloat64Array(): Float64Array
}

/**
 * Exposes the JavaScript
[DOMMatrix](https://developer.mozilla.org/en/docs/Web/API/DOMMatrix) to Kotlin
*/
public external open class DOMMatrix() : DOMMatrixReadOnly {
    constructor(transformList: String)
    constructor(other: DOMMatrixReadOnly)
    constructor(array32: Float32Array)
    constructor(array64: Float64Array)
    constructor(numberSequence: Array<Double>)
    override var a: Double
    override var b: Double
    override var c: Double
    override var d: Double
    override var e: Double
    override var f: Double
    override var m11: Double
    override var m12: Double
    override var m13: Double
    override var m14: Double
    override var m21: Double
    override var m22: Double
    override var m23: Double
    override var m24: Double
    override var m31: Double
    override var m32: Double
    override var m33: Double
    override var m34: Double
    override var m41: Double
    override var m42: Double
    override var m43: Double
    override var m44: Double
    override var is2D: Boolean
    override var isIdentity: Boolean
```
m21: Double
override var m22: Double
override var m23: Double
override var m24: Double
override var m31: Double
override var m32: Double
override var m33: Double
override var m34: Double
override var m41: Double
override var m42: Double
override var m43: Double
override var m44:

fun multiplySelf(other: DOMMatrix): DOMMatrix
fun preMultiplySelf(other: DOMMatrix): DOMMatrix
fun translateSelf(tx: Double, ty: Double, tz: Double = definedExternally): DOMMatrix
fun scaleSelf(scale: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
fun scaleNonUniformSelf(scaleX: Double, scaleY: Double = definedExternally, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
fun rotateSelf(angle: Double, originX: Double = definedExternally, originY: Double = definedExternally): DOMMatrix
fun rotateFromVectorSelf(x: Double, y: Double, z: Double, angle: Double): DOMMatrix
fun skewXSelf(sx: Double): DOMMatrix
fun skewYSelf(sy: Double): DOMMatrix
fun invertSelf(): DOMMatrix
fun setMatrixValue(transformList: String): DOMMatrix

public external interface ScrollOptions {
    var behavior: ScrollBehavior? /* = ScrollBehavior.AUTO */
    get() = definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ScrollOptions(behavior: ScrollBehavior? = ScrollBehavior.AUTO): ScrollOptions {
    val o = js("({})")
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [ScrollToOptions](https://developer.mozilla.org/en/docs/Web/API/ScrollToOptions) to Kotlin
 */
public external interface ScrollToOptions : ScrollOptions {
    var left: Double?
        get() = definedExternally
        set(value) = definedExternally
    var top: Double?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["left"] = left
    o["top"] = top
    o["behavior"] = behavior
    return o
}

/**
 * Exposes the JavaScript [MediaQueryList](https://developer.mozilla.org/en/docs/Web/API/MediaQueryList) to Kotlin
 */
public external abstract class MediaQueryList : EventTarget {
    open val media: String
    open val matches: Boolean
    open var onchange: ((Event) -> dynamic)?

    fun addListener(listener: EventListener?)
    fun addListener(listener: ((Event) -> Unit)?)
    fun removeListener(listener: EventListener?)
    fun removeListener(listener: ((Event) -> Unit)?)
}

/**
 * Exposes the JavaScript [MediaQueryListEvent](https://developer.mozilla.org/en/docs/Web/API/MediaQueryListEvent) to Kotlin
 */
public external open class MediaQueryListEvent(type: String, eventInitDict: MediaQueryListEventInit = definedExternally) : Event {
    open val media: String
    open val matches: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface MediaQueryListEventInit : EventInit {
    var media: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var matches: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["media"] = media
    o["matches"] = matches
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript [Screen](https://developer.mozilla.org/en/docs/Web/API/Screen) to Kotlin
 */
public external abstract class Screen {
    open val availWidth: Int
    open val availHeight: Int
    open val width: Int
    open val height: Int
    open val colorDepth: Int
}

/**
 * Exposes the JavaScript [CaretPosition](https://developer.mozilla.org/en/docs/Web/API/CaretPosition) to Kotlin
 */
public external abstract class CaretPosition {
    open val offsetNode: Node
    open val offset: Int
    fun getClientRect(): DOMRect?
}

public external interface ScrollIntoViewOptions : ScrollOptions {
    var block: ScrollLogicalPosition? /* = ScrollLogicalPosition.CENTER */
    get() = definedExternally
    set(value) = definedExternally
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2416
definedExternally
    set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun

    var box: BoxQuadOptions = BoxQuadOptions(
        var o = js(""(1)""))
    o["block"] = block
    o["inline"] = inline
    o["behavior"] = behavior
    return o
}

public external interface BoxQuadOptions {
    var box: CSSBoxType?
    var inline: CSSBoxType?
    var behavior: ScrollBehavior?
}

@kotlin.internal.InlineOnly
public inline fun BoxQuadOptions(box: CSSBoxType? = CSSBoxType.BORDER, inline: CSSBoxType? = CSSBoxType.BORDER, relativeTo: ScrollLogicalPosition = ScrollLogicalPosition.CENTER, behavior: ScrollBehavior = ScrollBehavior.AUTO): BoxQuadOptions {
    var box: DOMQuad = box
    var inline: DOMQuad = inline
    return box
}

public external interface ConvertCoordinateOptions {
    var fromBox: CSSBoxType?
    var toBox: CSSBoxType?
}

@kotlin.internal.InlineOnly
public inline fun ConvertCoordinateOptions(fromBox: CSSBoxType? = CSSBoxType.BORDER, toBox: CSSBoxType? = CSSBoxType.BORDER): ConvertCoordinateOptions {
    var fromBox: DOMQuad = fromBox
    var toBox: DOMQuad = toBox
    return fromBox
}

/**
 * Exposes the JavaScript
 * @see GeometryUtils
 */
public external interface GeometryUtils {
    fun getBoxQuads(options: BoxQuadOptions? = definedExternally): BoxQuadOptions
    fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions? = definedExternally): DOMQuad
    fun convertRectFromNode(rect: DOMRect, from: dynamic, options: ConvertCoordinateOptions? = definedExternally): DOMQuad
    fun convertPointFromNode(point: DOMPoint, from: dynamic, options: ConvertCoordinateOptions? = definedExternally): DOMPoint
}

/**
 * Exposes the JavaScript
 * @see Touch
 */
public external abstract class Touch {
    open val identifier: Int
    open val target: EventTarget
    open val screenX: Int
    open val screenY: Int
    open val clientX: Int
    open val clientY: Int
    open val pageX: Int
    open val pageY: Int
    open val region: String?
}

public external abstract class TouchList : ItemArrayLike<Touch> {
    override fun item(index: Int): Touch?
}

@kotlin.internal.InlineOnly
public inline operator fun TouchList.get(index: Int): Touch? = asDynamic()[index]

public external open class TouchEvent : UIEvent {
    open val touches: TouchList
    open val targetTouches: TouchList
    open val changedTouches: TouchList
    open val altKey: Boolean
    open val metaKey: Boolean
    open val ctrlKey: Boolean
    open val shiftKey: Boolean

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

/**
 * Exposes the JavaScript
 * @see Image
 */
public external open class Image(width: Int = definedExternally, height: Int = definedExternally) : HTMLImageElement {
    override var onabort: ((Event) -> dynamic)?
    override var onblur: ((FocusEvent) -> dynamic)?
    override var oncancel: ((Event) -> dynamic)?
    override var onchange: ((Event) -> dynamic)?
    override var onclick: ((MouseEvent) -> dynamic)?
    override var onclose: ((Event) -> dynamic)?
    override var oncontextmenu: ((MouseEvent) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
    override var oncontextmenu: ((Event) -> dynamic)?
}
dynamic)?\n override var onkeypress: ((KeyboardEvent) -> dynamic)?\n override var onkeyup:
((KeyboardEvent) -> dynamic)?\n override var onload: ((Event) -> dynamic)?\n override var onloadeddata:
((Event) -> dynamic)?\n override var onloadedmetadata: ((Event) -> dynamic)?\n override var onloadend:
((Event) -> dynamic)?\n override var onloadstart: ((ProgressEvent) -> dynamic)?\n override var onmousedown:
((MouseEvent) -> dynamic)?\n override var onmouseenter: ((MouseEvent) -> dynamic)?\n override var
onmouseleave: ((MouseEvent) -> dynamic)?\n override var onmousemove: ((MouseEvent) -> dynamic)?\n
override var onmouseout: ((MouseEvent) -> dynamic)?\n override var onmouseover: ((MouseEvent) ->
dynamic)?\n override var onmouseup: ((MouseEvent) -> dynamic)?\n override var onwheel: ((WheelEvent) ->
dynamic)?\n override var onpause: ((Event) -> dynamic)?\n override var onplay: ((Event) -> dynamic)?\n
override var onplaying: ((Event) -> dynamic)?\n override var onprogress: ((ProgressEvent) -> dynamic)?\n
override var onratechange: ((Event) -> dynamic)?\n override var onreset: ((Event) -> dynamic)?\n override var
onresize: ((Event) -> dynamic)?\n override var onscroll: ((Event) -> dynamic)?\n override var onseeked:
((Event) -> dynamic)?\n override var onseeking: ((Event) -> dynamic)?\n override var onselect: ((Event) ->
dynamic)?\n override var onshow: ((Event) -> dynamic)?\n override var onstalled: ((Event) -> dynamic)?\n
override var onsubmit: ((Event) -> dynamic)?\n override var onsuspend: ((Event) -> dynamic)?\n override var
ontimeupdate: ((Event) -> dynamic)?\n override var ontoggle: ((Event) -> dynamic)?\n override var
onvolumechange: ((Event) -> dynamic)?\n override var onwaiting: ((Event) -> dynamic)?\n override var
ongotpointercapture: ((PointerEvent) -> dynamic)?\n override var onlostpointercapture: ((PointerEvent) ->
dynamic)?\n override var onpointerdown: ((PointerEvent) -> dynamic)?\n override var onpointermove:
((PointerEvent) -> dynamic)?\n override var onpointerup: ((PointerEvent) -> dynamic)?\n override var
onpointercancel: ((PointerEvent) -> dynamic)?\n override var onpointerover: ((PointerEvent) -> dynamic)?\n
override var onpointerout: ((PointerEvent) -> dynamic)?\n override var onpointerenter: ((PointerEvent) ->
dynamic)?\n override var onpointerleave: ((PointerEvent) -> dynamic)?\n override var oncopy:
((ClipboardEvent) -> dynamic)?\n override var oncut: ((ClipboardEvent) -> dynamic)?\n override var onpaste:
((ClipboardEvent) -> dynamic)?\n override var contentEditable: String\n override val isContentEditable:
Boolean\n override val style: CSSStyleDeclaration\n override val children: HTMLCollection\n override val
firstElementChild: Element?\n override val lastElementChild: Element?\n override val childElementCount: Int\n
override val previousElementSibling: Element?\n override val nextElementSibling: Element?\n override val
assignedSlot: HTMLSlotElement?\n override fun prepend(vararg nodes: dynamic)\n override fun append(vararg
nodes: dynamic)\n override fun querySelector(selectors: String): Element?\n override fun
querySelectorAll(selectors: String): NodeList\n override fun before(vararg nodes: dynamic)\n override fun
after(vararg nodes: dynamic)\n override fun replaceWith(vararg nodes: dynamic)\n override fun remove()\n
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>\n override
fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMQuad\n override fun convertRectFromNode(rect: DOMRectReadOnly, from:
dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad\n override fun
convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* =
definedExternally */): DOMPoint\n\n companion object {\n
val ELEMENT_NODE: Short\n
val
ATTRIBUTE_NODE: Short\n
val TEXT_NODE: Short\n
val CDATA_SECTION_NODE: Short\n
val
ENTITY_REFERENCE_NODE: Short\n
val ENTITY_NODE: Short\n
val
PROCESSING_INSTRUCTION_NODE: Short\n
val COMMENT_NODE: Short\n
val
DOCUMENT_NODE: Short\n
val DOCUMENT_TYPE_NODE: Short\n
val
DOCUMENT_FRAGMENT_NODE: Short\n
val NOTATION_NODE: Short\n
val
DOCUMENT_POSITION_DISCONNECTED: Short\n
val DOCUMENT_POSITION_PRECEDING: Short\n
val DOCUMENT_POSITION_FOLLOWING: Short\n
val DOCUMENT_POSITION_CONTAINS: Short\n
val DOCUMENT_POSITION_CONTAINED_BY: Short\n
val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short\n }\n}\n\npublic external open class
Audio(src: String = definedExternally) : HTMLAudioElement {\n override var onabort: ((Event) -> dynamic)?\n

Open Source Used In IOx-FogDirector-AC18 1.17.0 2418


override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint

    companion object {
        val NETWORK_EMPTY: Short
        val NETWORK_IDLE: Short
        val NETWORK_LOADING: Short
        val NETWORK_NO_SOURCE: Short
        val HAVE_NOTHING: Short
        val HAVE_METADATA: Short
        val HAVE_CURRENT_DATA: Short
        val HAVE_FUTURE_DATA: Short
        val HAVE_ENOUGH_DATA: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }

/**
 * Exposes the JavaScript [Option](https://developer.mozilla.org/en/docs/Web/API/Option) to Kotlin
 */
public external open class Option(text: String = definedExternally, value: String = definedExternally, defaultSelected: Boolean = definedExternally, selected: Boolean = definedExternally) : HTMLOptionElement {
    override var onabort: ((Event) -> dynamic)?
    override var onblur: ((FocusEvent) -> dynamic)?
    override var oncancel: ((Event) -> dynamic)?
    override var oncanplay: ((Event) -> dynamic)?
    override var oncanplaythrough: ((Event) -> dynamic)?
    override var onchange: ((Event) -> dynamic)?
    override var onclick: ((MouseEvent) -> dynamic)?
    override var onclose: ((Event) -> dynamic)?
    override var oncontextmenu: ((MouseEvent) -> dynamic)?
    override var oncuechange: ((Event) -> dynamic)?
    override var ondblclick: ((MouseEvent) -> dynamic)?
    override var ondrag: ((DragEvent) -> dynamic)?
    override var ondragend: ((DragEvent) -> dynamic)?
    override var ondragenter: ((DragEvent) -> dynamic)?
    override var ondragexit: ((DragEvent) -> dynamic)?
    override var ondragleave: ((DragEvent) -> dynamic)?
    override var ondragover: ((DragEvent) -> dynamic)?
    override var ondragstart: ((DragEvent) -> dynamic)?
    override var ondurationchange: ((Event) -> dynamic)?
    override var onemptied: ((Event) -> dynamic)?
    override var onended: ((Event) -> dynamic)?
    override var onerror: ((dynamic, String, Int, Int, Any?) -> dynamic)?
    override var onfocus: ((FocusEvent) -> dynamic)?
    override var oninput: ((InputEvent) -> dynamic)?
    override var oninvalid: ((Event) -> dynamic)?
    override var onkeydown: ((KeyboardEvent) -> dynamic)?
    override var onkeypress: ((KeyboardEvent) -> dynamic)?
    override var onkeyup: ((KeyboardEvent) -> dynamic)?
    override var onload: ((Event) -> dynamic)?
    override var onloadeddata: ((Event) -> dynamic)?
    override var onloadedmetadata: ((Event) -> dynamic)?
    override var onloadedmetadata: ((Event) -> dynamic)?
    override var onlostpointercapture: ((PointerEvent) -> dynamic)?
    override var onmousedown: ((MouseEvent) -> dynamic)?
    override var onmouseenter: ((MouseEvent) -> dynamic)?
    override var onmouseleave: ((MouseEvent) -> dynamic)?
    override var onmousemove: ((MouseEvent) -> dynamic)?
    override var onmouseout: ((MouseEvent) -> dynamic)?
    override var onmouseover: ((MouseEvent) -> dynamic)?
    override var onmouseup: ((MouseEvent) -> dynamic)?
    override var onwheel: ((WheelEvent) -> dynamic)?
    override var onpause: ((Event) -> dynamic)?
    override var onplay: ((Event) -> dynamic)?
    override var onplaying: ((Event) -> dynamic)?
    override var onprogress: ((ProgressEvent) -> dynamic)?
    override var onratechange: ((Event) -> dynamic)?
    override var onreset: ((Event) -> dynamic)?
    override var onscroll: ((Event) -> dynamic)?
    override var onseeked: ((Event) -> dynamic)?
    override var onseeking: ((Event) -> dynamic)?
    override var onselect: ((Event) -> dynamic)?
    override var onshow: ((Event) -> dynamic)?
    override var onstalled: ((Event) -> dynamic)?
    override var onsubmit: ((Event) -> dynamic)?
    override var onvolumechange: ((Event) -> dynamic)?
    override var onwaiting: ((Event) -> dynamic)?
    override var ong otpointercapture: ((PointerEvent) -> dynamic)?
    override var onlostpointercapture: ((PointerEvent) -> dynamic)?
override var onpointerdown: ((PointerEvent) -> dynamic)?
override var onpointermove:
((PointerEvent) -> dynamic)?
override var onpointerup: ((PointerEvent) -> dynamic)?
override var onpointercancel: ((PointerEvent) -> dynamic)?
override var onpointerover: ((PointerEvent) -> dynamic)?
override var onpointerleave: ((PointerEvent) -> dynamic)?
override var oncopy:
((ClipboardEvent) -> dynamic)?
override var oncut: ((ClipboardEvent) -> dynamic)?
override var onpaste:
((ClipboardEvent) -> dynamic)?
override var oncontentEditable: String? override val isContentEditable:
override var onstyle: CSSStyleDeclaration? override val firstElementChild: Element?
override var onlastElementChild: Element?
override val childElementCount: Int?
override var onpreviousElementSibling: Element?
override var onnextElementSibling: Element?
override val querySelectorAll(selectors: String): NodeList
override fun prepend(vararg nodes: dynamic)
override fun append(vararg nodes: dynamic)
override fun querySelector(selectors: String): Element?
override fun querySelectorAll(selectors: String): NodeList
override fun after(vararg nodes: dynamic)
override fun before(vararg nodes: dynamic)
override fun remove()
override fun getBoxQuads(options: BoxQuadOptions /* = definedExternally */): Array<DOMQuad>
override fun convertQuadFromNode(quad: dynamic, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertRectFromNode(rect: DOMRectReadOnly, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMQuad
override fun convertPointFromNode(point: DOMPointInit, from: dynamic, options: ConvertCoordinateOptions /* = definedExternally */): DOMPoint

companion object {
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

public external interface UnionElementOrHTMLCollection

public external interface UnionElementOrRadioNodeList

public external interface UnionHTMLOptGroupElementOrHTMLOptionElement

public external interface UnionAudioTrackOrTextTrackOrVideoTrack

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface DocumentReadyState {
    companion object
}


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanPlayTypeResult {
    companion object
}

public inline val CanPlayTypeResult.Companion.EMPTY: CanPlayTypeResult get() = "".asDynamic().unsafeCast<CanPlayTypeResult>()

public inline val CanPlayTypeResult.Companion.MAYBE: CanPlayTypeResult get() = "maybe".asDynamic().unsafeCast<CanPlayTypeResult>()
CanvasLineJoin.get() = "bevel".asDynamic().unsafeCast<CanvasLineJoin>()
CanvasLineJoin.Companion.MITER = CanvasLineJoin.get() = "miter".asDynamic().unsafeCast<CanvasLineJoin>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextAlign {
    companion object
    public inline val CanvasTextAlign.Companion.MITER: CanvasTextAlign get() = "miter".asDynamic().unsafeCast<CanvasTextAlign>()
    public inline val CanvasTextAlign.Companion.START: CanvasTextAlign get() = "start".asDynamic().unsafeCast<CanvasTextAlign>()
    public inline val CanvasTextAlign.Companion.END: CanvasTextAlign get() = "end".asDynamic().unsafeCast<CanvasTextAlign>()
    public inline val CanvasTextAlign.Companion.LEFT: CanvasTextAlign get() = "left".asDynamic().unsafeCast<CanvasTextAlign>()
    public inline val CanvasTextAlign.Companion.RIGHT: CanvasTextAlign get() = "right".asDynamic().unsafeCast<CanvasTextAlign>()
    public inline val CanvasTextAlign.Companion.CENTER: CanvasTextAlign get() = "center".asDynamic().unsafeCast<CanvasTextAlign>()
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasTextBaseline {
    companion object
    public inline val CanvasTextBaseline.Companion.TOP: CanvasTextBaseline get() = "top".asDynamic().unsafeCast<CanvasTextBaseline>()
    public inline val CanvasTextBaseline.Companion.HANGING: CanvasTextBaseline get() = "hanging".asDynamic().unsafeCast<CanvasTextBaseline>()
    public inline val CanvasTextBaseline.Companion.MIDDLE: CanvasTextBaseline get() = "middle".asDynamic().unsafeCast<CanvasTextBaseline>()
    public inline val CanvasTextBaseline.Companion.ALPHABETIC: CanvasTextBaseline get() = "alphabetic".asDynamic().unsafeCast<CanvasTextBaseline>()
    public inline val CanvasTextBaseline.Companion.IDEOGRAPHIC: CanvasTextBaseline get() = "ideographic".asDynamic().unsafeCast<CanvasTextBaseline>()
    public inline val CanvasTextBaseline.Companion.BOTTOM: CanvasTextBaseline get() = "bottom".asDynamic().unsafeCast<CanvasTextBaseline>()
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface CanvasDirection {
    companion object
    public inline val CanvasDirection.Companion.LTR: CanvasDirection get() = "ltr".asDynamic().unsafeCast<CanvasDirection>()
    public inline val CanvasDirection.Companion.RTL: CanvasDirection get() = "rtl".asDynamic().unsafeCast<CanvasDirection>()
    public inline val CanvasDirection.Companion.INHERIT: CanvasDirection get() = "inherit".asDynamic().unsafeCast<CanvasDirection>()
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ScrollRestoration {
    companion object
    public inline val ScrollRestoration.Companion.AUTO: ScrollRestoration get() = "auto".asDynamic().unsafeCast<ScrollRestoration>()
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ImageOrientation {
    companion object
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface PremultiplyAlpha {
    companion object
    public inline val PremultiplyAlpha.Companion.NONE: PremultiplyAlpha get() = "none".asDynamic().unsafeCast<PremultiplyAlpha>()
    public inline val PremultiplyAlpha.Companion.PREMULTIPLY: PremultiplyAlpha get() = "premultiply".asDynamic().unsafeCast<PremultiplyAlpha>()
    public inline val PremultiplyAlpha.Companion.DEFAULT: PremultiplyAlpha get() = "default".asDynamic().unsafeCast<PremultiplyAlpha>()
}
ColorSpaceConversion {\n    companion object\n}\npublic inline val ColorSpaceConversion.Companion.NONE: ColorSpaceConversion get() = "none".asDynamic().unsafeCast<ColorSpaceConversion>()\n
public inline val ColorSpaceConversion.Companion.DEFAULT: ColorSpaceConversion get() = "default".asDynamic().unsafeCast<ColorSpaceConversion>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface ResizeQuality {
    companion object\n}\npublic inline val ResizeQuality.Companion.PIXELATED: ResizeQuality get() = "pixelated".asDynamic().unsafeCast<ResizeQuality>()\n
public inline val ResizeQuality.Companion.LOW: ResizeQuality get() = "low".asDynamic().unsafeCast<ResizeQuality>()\n
public inline val ResizeQuality.Companion.MEDIUM: ResizeQuality get() = "medium".asDynamic().unsafeCast<ResizeQuality>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface BinaryType {
    companion object\n}\npublic inline val BinaryType.Companion.BLOB: BinaryType get() = "blob".asDynamic().unsafeCast<BinaryType>()\n
public inline val BinaryType.Companion.ARRAYBUFFER: BinaryType get() = "arraybuffer".asDynamic().unsafeCast<BinaryType>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface WorkerType {
    companion object\n}\npublic inline val WorkerType.Companion.CLASSIC: WorkerType get() = "classic".asDynamic().unsafeCast<WorkerType>()\n
public inline val WorkerType.Companion.MODULE: WorkerType get() = "module".asDynamic().unsafeCast<WorkerType>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface ShadowRootMode {
    companion object\n}\npublic inline val ShadowRootMode.Companion.OPEN: ShadowRootMode get() = "open".asDynamic().unsafeCast<ShadowRootMode>()\n
public inline val ShadowRootMode.Companion.CLOSED: ShadowRootMode get() = "closed".asDynamic().unsafeCast<ShadowRootMode>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface ScrollBehavior {
    companion object\n}\npublic inline val ScrollBehavior.Companion.AUTO: ScrollBehavior get() = "auto".asDynamic().unsafeCast<ScrollBehavior>()\n
public inline val ScrollBehavior.Companion.INSTANT: ScrollBehavior get() = "instant".asDynamic().unsafeCast<ScrollBehavior>()\n
public inline val ScrollBehavior.Companion.SMOOTH: ScrollBehavior get() = "smooth".asDynamic().unsafeCast<ScrollBehavior>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface ScrollLogicalPosition {
    companion object\n}\npublic inline val ScrollLogicalPosition.Companion.START: ScrollLogicalPosition get() = "start".asDynamic().unsafeCast<ScrollLogicalPosition>()\n
public inline val ScrollLogicalPosition.Companion.CENTER: ScrollLogicalPosition get() = "center".asDynamic().unsafeCast<ScrollLogicalPosition>()\n
public inline val ScrollLogicalPosition.Companion.END: ScrollLogicalPosition get() = "end".asDynamic().unsafeCast<ScrollLogicalPosition>()\n
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")\npublic external interface CSSBoxType {
    companion object\n}\npublic inline val CSSBoxType.Companion.MARGIN: CSSBoxType get() = "margin".asDynamic().unsafeCast<CSSBoxType>()\n
public inline val CSSBoxType.Companion.BORDER: CSSBoxType get() = "border".asDynamic().unsafeCast<CSSBoxType>()\n
public inline val CSSBoxType.Companion.CONTENT: CSSBoxType get() = "content".asDynamic().unsafeCast<CSSBoxType>()\n
/* * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

Open Source Used In IOx-FogDirector-AC18 1.1.7.0 2424
DO NOT EDIT!// See github.com/kotlin/dukat for details

package org.w3c.fetch

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.
import org.w3c.dom.*
import org.w3c.dom.clipboard.
import org.w3c.dom.css.
import org.w3c.dom.events.
import org.w3c.dom.media.
import org.w3c.dom.parsing.
import org.w3c.dom.pointer.
import org.w3c.dom.svg.
import org.w3c.dom.url.
import org.w3c.files.
import org.w3c.notifications.
import org.w3c.performance.
import org.w3c.workers.
import org.w3c.xhr.

/**
 * Exposes the JavaScript [Headers](https://developer.mozilla.org/en/docs/Web/API/Headers) to Kotlin
 */
public external open class Headers(init: dynamic = definedExternally) {
    fun append(name: String, value: String)
    fun delete(name: String)
    fun get(name: String): String?
    fun has(name: String): Boolean
    fun set(name: String, value: String)
}

/**
 * Exposes the JavaScript [Body](https://developer.mozilla.org/en/docs/Web/API/Body) to Kotlin
 */
public external interface Body {
    val bodyUsed: Boolean
    fun arrayBuffer(): Promise<ArrayBuffer>
    fun blob(): Promise<Blob>
    fun formData(): Promise<FormData>
    fun json(): Promise<String>
}

/**
 * Exposes the JavaScript [Request](https://developer.mozilla.org/en/docs/Web/API/Request) to Kotlin
 */
public external open class Request(input: dynamic, init: RequestInit = definedExternally) : Body {
    open val method: String
    open val url: String
    open val headers: Headers
    open val type: RequestType
    open val destination:
    open val referrer: String
    open val referrerPolicy: dynamic
    open val mode:
    open val credentials: RequestCredentials
    open val cache:
    open val redirect:
    open val integrity: String
    open val keepalive: Boolean
    override val bodyUsed: Boolean
    fun clone(): Request
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun formData(): Promise<FormData>
    override fun json(): Promise<String>
}

public external interface RequestInit {
    var method: String?
    var headers: dynamic
    var body: dynamic
    var referrer: String?
    var referrerPolicy: dynamic
    var mode: RequestMode?
    var credentials: RequestCredentials?
    var cache: RequestCache?
    var redirect: RequestRedirect?
    var integrity: String?
    var keepalive: Boolean?
    var window:
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["method"] = method
    o["headers"] = headers
    o["body"] = body
    o["referrer"] = referrer
    o["referrerPolicy"] = referrerPolicy
    o["mode"] = mode
    o["credentials"] = credentials
    o["cache"] = cache
    o["redirect"] = redirect
    o["integrity"] = integrity
    o["keepalive"] = keepalive
    o["window"] = window
    return o
}

/**
 * Exposes the JavaScript [Response](https://developer.mozilla.org/en/docs/Web/API/Response) to Kotlin
 */
public external open class Response(body: dynamic = definedExternally, init: ResponseInit = definedExternally) : Body {
    open val type: ResponseType
    open val url: String
    open val redirected: Boolean
    open val status: Short
    open val ok: Boolean
    open val statusText: String
    open val headers: Headers
    open val body: dynamic
    open val trailer: Promise<Headers>
    override val bodyUsed: Boolean
    fun clone(): Response
    override fun arrayBuffer(): Promise<ArrayBuffer>
    override fun blob(): Promise<Blob>
    override fun formData(): Promise<FormData>
    override fun json(): Promise<String>
}
override fun text(): Promise<String>

companion object {
    fun error(): Response
    fun redirect(url: String, status: Short = definedExternally): Response
}

public external interface ResponseInit {
    var status: Short? /* = 200 */
        get() = definedExternally
        set(value) = definedExternally
    var statusText: String? /* = "OK" */
        get() = definedExternally
        set(value) = definedExternally
    var headers: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["status"] = status
    o["statusText"] = statusText
    o["headers"] = headers
    return o
}

public inline val RequestType.Companion.EMPTY: RequestType get() = "".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.AUDIO: RequestType get() = "audio".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.FONT: RequestType get() = "font".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.IMAGE: RequestType get() = "image".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.SCRIPT: RequestType get() = "script".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.STYLE: RequestType get() = "style".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.TRACK: RequestType get() = "track".asDynamic().unsafeCast<RequestType>()

public inline val RequestType.Companion.VIDEO: RequestType get() = "video".asDynamic().unsafeCast<RequestType>()

public inline val RequestDestination.Companion.EMPTY: RequestDestination get() = "".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.DOCUMENT: RequestDestination get() = "document".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.EMBED: RequestDestination get() = "embed".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.FONT: RequestDestination get() = "font".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.IMAGE: RequestDestination get() = "image".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.MANIFEST: RequestDestination get() = "manifest".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.MEDIA: RequestDestination get() = "media".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.OBJECT: RequestDestination get() = "object".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.REPORT: RequestDestination get() = "report".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SERVICEWORKER: RequestDestination get() = "serviceworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.SHAREDWORKER: RequestDestination get() = "sharedworker".asDynamic().unsafeCast<RequestDestination>()

public inline val RequestDestination.Companion.STYLE: RequestDestination get() = "style".asDynamic().unsafeCast<RequestDestination>()
/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestMode {
    companion object
}

public inline val RequestMode.Companion.NAVIGATE: RequestMode get() = "navigate".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.SAME_ORIGIN: RequestMode get() = "same-origin".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.NO_CORS: RequestMode get() = "no-cors".asDynamic().unsafeCast<RequestMode>()

public inline val RequestMode.Companion.CORS: RequestMode get() = "cors".asDynamic().unsafeCast<RequestMode>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCredentials {
    companion object
}

public inline val RequestCredentials.Companion.OMIT: RequestCredentials get() = "omit".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.SAME_ORIGIN: RequestCredentials get() = "same-origin".asDynamic().unsafeCast<RequestCredentials>()

public inline val RequestCredentials.Companion.INCLUDE: RequestCredentials get() = "include".asDynamic().unsafeCast<RequestCredentials>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestCache {
    companion object
}

public inline val RequestCache.Companion.DEFAULT: RequestCache get() = "default".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_STORE: RequestCache get() = "no-store".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.RELOAD: RequestCache get() = "replay".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.NO_CACHE: RequestCache get() = "no-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.FORCE_CACHE: RequestCache get() = "force-cache".asDynamic().unsafeCast<RequestCache>()

public inline val RequestCache.Companion.ONLY_IF_CACHED: RequestCache get() = "only-if-cached".asDynamic().unsafeCast<RequestCache>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface RequestRedirect {
    companion object
}

public inline val RequestRedirect.Companion.FOLLOW: RequestRedirect get() = "follow".asDynamic().unsafeCast<RequestRedirect>()

public inline val RequestRedirect.Companion.ERROR: RequestRedirect get() = "error".asDynamic().unsafeCast<RequestRedirect>()

public inline val RequestRedirect.Companion.MANUAL: RequestRedirect get() = "manual".asDynamic().unsafeCast<RequestRedirect>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ResponseType {
    companion object
}

public inline val ResponseType.Companion.BASIC: ResponseType get() = "basic".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.CORS: ResponseType get() = "cors".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.DEFAULT: ResponseType get() = "default".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.ERROR: ResponseType get() = "error".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.OPAQUE: ResponseType get() = "opaque".asDynamic().unsafeCast<ResponseType>()

public inline val ResponseType.Companion.OPAQUEREDIRECT: ResponseType get() = "opaqueredirect".asDynamic().unsafeCast<ResponseType>()

/* * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that
Open Source Used In IOx-FogDirector-AC18 1.17.0 2428

can be found in the license/LICENSE.txt file. 

NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! 

See github.com/kotlin/dukat for details 

package org.w3c.dom.mediacapture 

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [MediaStream](https://developer.mozilla.org/en/docs/Web/API/MediaStream) to Kotlin
 */
public external open class MediaStream() : EventTarget {
    constructor(stream: MediaStream)
    constructor(tracks: Array<MediaStreamTrack>)
    open val id: String
    open val active: Boolean
    var onaddtrack: ((MediaStreamTrackEvent) -> dynamic)?
    var onremovetrack: ((MediaStreamTrackEvent) -> dynamic)?
    fun getAudioTracks(): Array<MediaStreamTrack>
    fun getVideoTracks(): Array<MediaStreamTrack>
    fun getTracks(): Array<MediaStreamTrack>
    fun getTrackById(trackId: String): MediaStreamTrack?
    fun addTrack(track: MediaStreamTrack)
    fun removeTrack(track: MediaStreamTrack)
    fun clone(): MediaStream
}

/**
 * Exposes the JavaScript [MediaStreamTrack](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrack) to Kotlin
 */
public external abstract class MediaStreamTrack : EventTarget {
    open val kind: String
    open val id: String
    open val label: String
    open var enabled: Boolean
    open val muted: Boolean
    open var onmute: ((Event) -> dynamic)?
    open var onunmute: ((Event) -> dynamic)?
    open val readyState: MediaStreamTrackState
    open var onended: ((Event) -> dynamic)?
    open var onoverconstrained: ((Event) -> dynamic)?
    fun clone(): MediaStreamTrack
    fun stop()
    fun getCapabilities(): MediaTrackCapabilities
    fun getConstraints(): MediaTrackSupportedConstraints
    fun applyConstraints(constraints: MediaTrackSupportedConstraints = definedExternally): Promise<Unit>
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeModel"] = resizeMode
    o["volume"] = volume

o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
        return o
}
public external interface MediaTrackCapabilities {
    var width: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var height: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: Array<String>?
        get() = definedExternally
        set(value) = definedExternally
    var volume: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: Array<false>?
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: Array<false>?
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: Array<false>?
        get() = definedExternally
        set(value) = definedExternally
    var latency: DoubleRange?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: ULongRange?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
        o["width"] = width
        o["height"] = height
        o["aspectRatio"] = aspectRatio
        o["frameRate"] = frameRate
        o["facingMode"] = facingMode
        o["resizeMode"] = resizeMode
        o["sampleRate"] = sampleRate
        o["sampleSize"] = sampleSize
        o["echoCancellation"] = echoCancellation
        o["autoGainControl"] = autoGainControl
        o["noiseSuppression"] = noiseSuppression
        o["latency"] = latency
        o["channelCount"] = channelCount
        o["deviceId"] = deviceId
        o["groupId"] = groupId
            return o
}

/**
 * Exposes the JavaScript [MediaTrackConstraints](https://developer.mozilla.org/en/docs/Web/API/MediaTrackConstraints) to Kotlin
 */
public external interface MediaTrackConstraints : MediaTrackConstraintSet {
    var advanced: Array<MediaTrackConstraintSet>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
        o["width"] = width
        o["height"] = height
        o["aspectRatio"] = aspectRatio
        o["frameRate"] = frameRate
        o["facingMode"] = facingMode
        o["resizeMode"] = resizeMode
        o["volume"] = volume
        o["sampleRate"] = sampleRate
        o["sampleSize"] = sampleSize
        o["echoCancellation"] = echoCancellation
        o["autoGainControl"] = autoGainControl
        o["noiseSuppression"] = noiseSuppression
        o["latency"] = latency
        o["channelCount"] = channelCount
        o["deviceId"] = deviceId
        o["groupId"] = groupId
            return o
}
MediaTrackConstraintSet {
    var width: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var height: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var volume: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var latency: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var groupId: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["width"] = width
    o["height"] = height
    o["aspectRatio"] = aspectRatio
    o["frameRate"] = frameRate
    o["facingMode"] = facingMode
    o["resizeMode"] = resizeMode
    o["volume"] = volume
    o["sampleRate"] = sampleRate
    o["sampleSize"] = sampleSize
    o["echoCancellation"] = echoCancellation
    o["autoGainControl"] = autoGainControl
    o["noiseSuppression"] = noiseSuppression
    o["latency"] = latency
    o["channelCount"] = channelCount
    o["deviceId"] = deviceId
    o["groupId"] = groupId
    return o
}

/**
 * Exposes the JavaScript
 * [MediaTrackSettings](https://developer.mozilla.org/en/docs/Web/API/MediaTrackSettings) to Kotlin
 */
public external interface MediaTrackSettings {
    var width: Int?
        get() = definedExternally
        set(value) = definedExternally
    var height: Int?
        get() = definedExternally
        set(value) = definedExternally
    var aspectRatio: Double?
        get() = definedExternally
        set(value) = definedExternally
    var frameRate: Double?
        get() = definedExternally
        set(value) = definedExternally
    var facingMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var resizeMode: String?
        get() = definedExternally
        set(value) = definedExternally
    var volume: Double?
        get() = definedExternally
        set(value) = definedExternally
    var sampleRate: Int?
        get() = definedExternally
        set(value) = definedExternally
    var sampleSize: Int?
        get() = definedExternally
        set(value) = definedExternally
    var echoCancellation: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var autoGainControl: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var noiseSuppression: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var latency: Double?
        get() = definedExternally
        set(value) = definedExternally
    var channelCount: Int?
        get() = definedExternally
        set(value) = definedExternally
    var deviceId: String?
        get() = definedExternally
        set(value) = definedExternally
    var groupId: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
o["width"] = width
  o["height"] = height
  o["aspectRatio"] = aspectRatio
  o["frameRate"] = frameRate
  o["facingMode"] = facingMode
  o["resizeMode"] = resizeMode
  o["volume"] = volume
  o["sampleRate"] = sampleRate
  o["sampleSize"] = sampleSize
  o["echoCancellation"] = echoCancellation
  o["autoGainControl"] = autoGainControl
  o["noiseSuppression"] = noiseSuppression
  o["latency"] = latency
  o["channelCount"] = channelCount
  o["deviceId"] = deviceId
  o["groupId"] = groupId
  return o
}

/**
 * Exposes the JavaScript
 [MediaStreamTrackEvent](https://developer.mozilla.org/en/docs/Web/API/MediaStreamTrackEvent) to Kotlin
 */
public external open class MediaStreamTrackEvent(type: String, eventInitDict: MediaStreamTrackEventInit) : Event {
    open val track: MediaStreamTrack

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface MediaStreamTrackEventInit : EventInit {
        var track: MediaStreamTrack?
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["track"] = track
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    public external open class OverconstrainedErrorEvent(type: String, eventInitDict: OverconstrainedErrorEventInit) : Event {
        open val error: dynamic

        companion object {
            val NONE: Short
            val CAPTURING_PHASE: Short
            val AT_TARGET: Short
            val BUBBLING_PHASE: Short
        }

        public external interface OverconstrainedErrorEventInit : EventInit {
            var error: dynamic /* = null */
                get() = definedExternally
                set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["error"] = error
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /**
     * Exposes the JavaScript
     [MediaDevices](https://developer.mozilla.org/en/docs/Web/API/MediaDevices) to Kotlin
     */
    public external abstract class MediaDevices : EventTarget {
        open var ondevicechange: ((Event) -> dynamic)?
        fun enumerateDevices(): Promise<Array<MediaDeviceInfo>>
        fun getSupportedConstraints(): MediaTrackSupportedConstraints
        fun getUserMedia(constraints: MediaStreamConstraints = definedExternally): Promise<MediaStream>
    }

    /**
     * Exposes the JavaScript
     [MediaDeviceInfo](https://developer.mozilla.org/en/docs/Web/API/MediaDeviceInfo) to Kotlin
     */
    public external abstract class MediaDeviceInfo {
        open val deviceId: String
        open val kind: MediaDeviceKind
        open val label: String
        fun toJSON(): dynamic
    }

    public external abstract class InputDeviceInfo : MediaDeviceInfo {
        fun getCapabilities(): MediaTrackCapabilities
    }

    /**
     * Exposes the JavaScript
     [DoubleRange](https://developer.mozilla.org/en/docs/Web/API/DoubleRange) to Kotlin
     */
    public external interface DoubleRange {
        var max: Double?
            get() = definedExternally
            set(value) = definedExternally
        var min: Double?
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
    public inline fun DoubleRange(max: Double? = undefined, min: Double? = undefined): DoubleRange {
        val o = js("({})")
        o["max"] = max
        o["min"] = min
        return o
    }
public external interface ConstrainDoubleRange : DoubleRange {
    var exact: Double?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Double?
        get() = definedExternally
        set(value) = definedExternally
    var max: Number?
        get() = definedExternally
        set(value) = definedExternally
    var min: Number?
        get() = definedExternally
        set(value) = definedExternally
}

public external interface ConstrainDOMStringParameters {
    var exact: dynamic
        get() = definedExternally
        set(value) = definedExternally
    var ideal: dynamic
        get() = definedExternally
        set(value) = definedExternally
}

public external interface ConstrainBooleanParameters {
    var exact: Boolean?
        get() = definedExternally
        set(value) = definedExternally
    var ideal: Boolean?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun ConstrainDOMStringParameters(exact: dynamic = undefined, ideal: dynamic = undefined): ConstrainDOMStringParameters {
    val o = js("{"")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}

@kotlin.internal.InlineOnly
    val o = js("{"")
    o["exact"] = exact
    o["ideal"] = ideal
    return o
}

/**
 * Exposes the JavaScript
[ConstrainDOMStringParameters](https://developer.mozilla.org/en/docs/Web/API/ConstrainDOMStringParameters) to Kotlin
 */

@kotlin.internal.InlineOnly
public inline fun Capabilities(): Capabilities {
    val o = js("{"")
    return o
}

@kotlin.internal.InlineOnly
public inline fun Settings(): Settings {
    val o = js("{"")
    return o
}

@kotlin.internal.InlineOnly
public inline fun ConstraintSet(): ConstraintSet {
    val o = js("{"")
    return o
}

@kotlin.internal.InlineOnly
public inline fun Constraints(advanced: Array<ConstraintSet>? = undefined): Constraints {
    val o = js("{"")
    o["advanced"] = advanced
    return o
}

/* please, don't implement this interface!
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaStreamTrackState {
    companion object
}

public inline val MediaStreamTrackState.Companion.LIVE: MediaStreamTrackState get() = "live".asDynamic().unsafeCast<MediaStreamTrackState>()

public inline val MediaStreamTrackState.Companion.ENDED: MediaStreamTrackState get() = "ended".asDynamic().unsafeCast<MediaStreamTrackState>()

/* please, don't implement this interface!
*/
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoFacingModeEnum {
    companion object
}

public inline val VideoFacingModeEnum.Companion.USER: VideoFacingModeEnum get() = "user".asDynamic().unsafeCast<VideoFacingModeEnum>()

VideoFacingModeEnum.Companion.LEFT: VideoFacingModeEnum get() = "left".asDynamic().unsafeCast<VideoFacingModeEnum>()

VideoFacingModeEnum.Companion.RIGHT: VideoFacingModeEnum get() = "right".asDynamic().unsafeCast<VideoFacingModeEnum>()

VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = "none".asDynamic().unsafeCast<VideoResizeModeEnum>()

VideoResizeModeEnum.Companion.CROP_AND_SCALE: VideoResizeModeEnum get() = "crop-and-scale".asDynamic().unsafeCast<VideoResizeModeEnum>()

MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface VideoResizeModeEnum {
    companion object
}

public inline val VideoResizeModeEnum.Companion.NONE: VideoResizeModeEnum get() = "none".asDynamic().unsafeCast<VideoResizeModeEnum>()


/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface MediaDeviceKind {
    companion object
}

public inline val MediaDeviceKind.Companion.AUDIOINPUT: MediaDeviceKind get() = "audioinput".asDynamic().unsafeCast<MediaDeviceKind>()

public inline val MediaDeviceKind.Companion.AUDIOOUTPUT: MediaDeviceKind get() = "audiooutput".asDynamic().unsafeCast<MediaDeviceKind>()

--

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.dom.pointerevents

public external interface PointerEventInit : MouseEventInit {
    var pointerId: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var width: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var height: Double? /* = 1.0 */
        get() = definedExternally
        set(value) = definedExternally
    var pressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tangentialPressure: Float? /* = 0f */
        get() = definedExternally
        set(value) = definedExternally
    var tiltX: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally

    var tiltY: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var twist: Int? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var pointerType: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
    var isPrimary: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly

get() = definedExternally

set(value) = definedExternally

var isPrimary: Boolean? = false

window: Window? = null,

detail: Int? = 0,
bubbles: Boolean? = false,
cancelable: Boolean? = false,
composed: Boolean? = false,
pointerEventId: Int? = 0,
width: Double? = 1.0,
height: Double? = 1.0,
pressure: Float? = 0f,
tangentialPressure: Float? = 0f,
tiltX: Int? = 0,
tiltY: Int? = 0,
val ELEMENT_NODE: Short
val ATTRIBUTE_NODE: Short
val TEXT_NODE: Short
val CDATA_SECTION_NODE: Short
val ENTITY_REFERENCE_NODE: Short
val ENTITY_NODE: Short
val PROCESSING_INSTRUCTION_NODE: Short
val COMMENT_NODE: Short
val DOCUMENT_NODE: Short
val DOCUMENT_TYPE_NODE: Short
val DOCUMENT_FRAGMENT_NODE: Short
val DOCUMENT_POSITION_DISCONNECTED: Short
val DOCUMENT_POSITION_PRECEDING: Short
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short
val DOCUMENT_POSITION_CONTAINED_BY: Short

/**
 * Exposes the JavaScript [SVGGeometryElement](https://developer.mozilla.org/en/docs/Web/API/SVGGeometryElement) to Kotlin
 */
public external abstract class SVGGeometryElement : SVGGraphicsElement {
    open val pathLength: SVGAnimatedNumber
    fun isPointInFill(point: DOMPoint): Boolean
    fun isPointInStroke(point: DOMPoint): Boolean
    fun getTotalLength(): Float
    fun getPointAtLength(distance: Float): DOMPoint

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

/**
 * Exposes the JavaScript [SVGNumber](https://developer.mozilla.org/en/docs/Web/API/SVGNumber) to Kotlin
 */
public external abstract class SVGNumber {
    open var value: Float
}

/**
 * Exposes the JavaScript [SVGLength](https://developer.mozilla.org/en/docs/Web/API/SVGLength) to Kotlin
 */
public external abstract class SVGLength {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_LENGTHTYPE_UNKNOWN: Short
        val SVG_LENGTHTYPE_NUMBER: Short
        val SVG_LENGTHTYPE_PERCENTAGE: Short
        val SVG_LENGTHTYPE_EMS: Short
        val SVG_LENGTHTYPE_EXS: Short
        val SVG_LENGTHTYPE_PX: Short
        val SVG_LENGTHTYPE_CM: Short
        val SVG_LENGTHTYPE_MM: Short
        val SVG_LENGTHTYPE_IN: Short
        val SVG_LENGTHTYPE_PT: Short
        val SVG_LENGTHTYPE_PC: Short
    }
}

/**
 * Exposes the JavaScript [SVGAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAngle) to Kotlin
 */
public external abstract class SVGAngle {
    open val unitType: Short
    open var value: Float
    open var valueInSpecifiedUnits: Float
    open var valueAsString: String
    fun newValueSpecifiedUnits(unitType: Short, valueInSpecifiedUnits: Float)
    fun convertToSpecifiedUnits(unitType: Short)

    companion object {
        val SVG_ANGLETYPE_UNKNOWN: Short
        val SVG_ANGLETYPE_UNSPECIFIED: Short
        val SVG_ANGLETYPE_DEG: Short
        val SVG_ANGLETYPE_RAD: Short
        val SVG_ANGLETYPE_GRAD: Short
    }
}

public external abstract class SVGNameList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: dynamic): dynamic
    fun insertItemBefore(newItem: dynamic, index: Int): dynamic
    fun replaceItem(newItem: dynamic, index: Int): dynamic
    fun removeItem(index: Int): dynamic
    fun appendItem(newItem: dynamic): dynamic
    fun getItem(index: Int): dynamic
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.get(index: Int): dynamic = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNameList.set(index: Int, newItem: dynamic) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGNumberList) to Kotlin
 */
Kotlin

```kotlin
public external abstract class SVGNumberList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGNumber): SVGNumber
    fun insertItemBefore(newItem: SVGNumber, index: Int): SVGNumber
    fun replaceItem(newItem: SVGNumber, index: Int): SVGNumber
    fun removeItem(index: Int): SVGNumber
    fun appendItem(newItem: SVGNumber): SVGNumber
    fun getItem(index: Int): SVGNumber
}

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.get(index: Int): SVGNumber? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGNumberList.set(index: Int, newItem: SVGNumber) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin
 */
public external abstract class SVGLengthList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: SVGLength): SVGLength
    fun insertItemBefore(newItem: SVGLength, index: Int): SVGLength
    fun replaceItem(newItem: SVGLength, index: Int): SVGLength
    fun removeItem(index: Int): SVGLength
    fun appendItem(newItem: SVGLength): SVGLength
    fun getItem(index: Int): SVGLength
}

@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.get(index: Int): SVGLength? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGLengthList.set(index: Int, newItem: SVGLength) { asDynamic()[index] = newItem }

/**
 * Exposes the JavaScript [SVGAnimatedBoolean](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedBoolean) to Kotlin
 */
public external abstract class SVGAnimatedBoolean {
    open var baseVal: Boolean
    open val animVal: Boolean
}

/**
 * Exposes the JavaScript [SVGAnimatedEnumeration](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedEnumeration) to Kotlin
 */
public external abstract class SVGAnimatedEnumeration {
    open var baseVal: Short
    open val animVal: Short
}

/**
 * Exposes the JavaScript [SVGAnimatedInteger](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedInteger) to Kotlin
 */
public external abstract class SVGAnimatedInteger {
    open var baseVal: Int
    open val animVal: Int
}

/**
 * Exposes the JavaScript [SVGAnimatedNumber](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumber) to Kotlin
 */
public external abstract class SVGAnimatedNumber {
    open var baseVal: Float
    open val animVal: Float
}

/**
 * Exposes the JavaScript [SVGAnimatedLength](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedLength) to Kotlin
 */
public external abstract class SVGAnimatedLength {
    open val baseVal: SVGLength
    open val animVal: SVGLength
}

/**
 * Exposes the JavaScript [SVGAnimatedAngle](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedAngle) to Kotlin
 */
public external abstract class SVGAnimatedAngle {
    open val baseVal: SVGAngle
    open val animVal: SVGAngle
}

/**
 * Exposes the JavaScript [SVGAnimatedString](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedString) to Kotlin
 */
public external abstract class SVGAnimatedString {
    open var baseVal: String
    open val animVal: String
}

/**
 * Exposes the JavaScript [SVGAnimatedRect](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedRect) to Kotlin
 */
public external abstract class SVGAnimatedRect {
    open val baseVal: DOMRect
    open val animVal: DOMRect
}

/**
 * Exposes the JavaScript [SVGAnimatedNumberList](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedNumberList) to Kotlin
 */
public external abstract class SVGAnimatedNumberList {
    open val baseVal: SVGNumberList
    open val animVal: SVGNumberList
}

/**
 * Exposes the JavaScript [SVGLengthList](https://developer.mozilla.org/en/docs/Web/API/SVGLengthList) to Kotlin
 */
public external abstract class SVGLengthList {
    open val baseVal: SVGLengthList
    open val animVal: SVGLengthList
}
```

Open Source Used In IOx-FogDirector-AC18 1.17.0 2436
Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGUnknownElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGDefsElement](https://developer.mozilla.org/en/docs/Web/API/SVGDefsElement) to Kotlin
 */
public external abstract class SVGDefsElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGDescElement](https://developer.mozilla.org/en/docs/Web/API/SVGDescElement) to Kotlin
 */
public external abstract class SVGDescElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGMetadataElement](https://developer.mozilla.org/en/docs/Web/API/SVGMetadataElement) to Kotlin
 */
public external abstract class SVGMetadataElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
** Exposes the JavaScript [SVGTitleElement](https://developer.mozilla.org/en/docs/Web/API/SVGTitleElement) to Kotlin

```kotlin
public external abstract class SVGTitleElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

** Exposes the JavaScript [SVGSymbolElement](https://developer.mozilla.org/en/docs/Web/API/SVGSymbolElement) to Kotlin

```kotlin
public external abstract class SVGSymbolElement : SVGGraphicsElement, SVGFitToViewBox {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

** Exposes the JavaScript [SVGUseElement](https://developer.mozilla.org/en/docs/Web/API/SVGUseElement) to Kotlin

```kotlin
public external abstract class SVGUseElement : SVGGraphicsElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val instanceRoot: SVGElement?
    open val animatedInstanceRoot: SVGElement?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```

```kotlin
public external open class SVGUseElementShadowRoot : ShadowRoot {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
```
class SVGElementInstance {
    val correspondingElement: SVGElement?
        get() = definedExternally
    val correspondingUseElement: SVGUseElement?
        get() = definedExternally
}

public external open class ShadowAnimation(source: dynamic, newTarget: dynamic) {
    open val sourceAnimation: dynamic
}

/**
 * Exposes the JavaScript [SVGSwitchElement](https://developer.mozilla.org/en/docs/Web/API/SVGSwitchElement) to Kotlin
 */
public external abstract class SVGSwitchElement : SVGGraphicsElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }

    public external interface GetSVGDocument {
        fun getSVGDocument(): Document
    }

    /**
     * Exposes the JavaScript [SVGStyleElement](https://developer.mozilla.org/en/docs/Web/API/SVGStyleElement) to Kotlin
     */
    public external abstract class SVGStyleElement : SVGElement, LinkStyle {
        open var type: String
        open var media: String
        open var title: String

        companion object {
            val ELEMENT_NODE: Short
            val ATTRIBUTE_NODE: Short
            val TEXT_NODE: Short
            val CDATA_SECTION_NODE: Short
            val ENTITY_REFERENCE_NODE: Short
            val ENTITY_NODE: Short
            val PROCESSING_INSTRUCTION_NODE: Short
            val COMMENT_NODE: Short
            val DOCUMENT_NODE: Short
            val DOCUMENT_TYPE_NODE: Short
            val DOCUMENT_FRAGMENT_NODE: Short
            val NOTATION_NODE: Short
            val DOCUMENT_POSITION_DISCONNECTED: Short
            val DOCUMENT_POSITION_PRECEDING: Short
            val DOCUMENT_POSITION_FOLLOWING: Short
            val DOCUMENT_POSITION_CONTAINS: Short
            val DOCUMENT_POSITION_CONTAINED_BY: Short
        }

        /**
         * Exposes the JavaScript [SVGTransform](https://developer.mozilla.org/en/docs/Web/API/SVGTransform) to Kotlin
         */
        public external abstract class SVGTransform {
            open val type: Short
            open val matrix: DOMMatrix
            open val angle: Float
            fun setMatrix(matrix: DOMMatrixReadOnly)
            fun setTranslate(tx: Float, ty: Float)
            fun setScale(sx: Float, sy: Float)
            fun setRotate(angle: Float, cx: Float, cy: Float)
            fun setSkewX(angle: Float)
            fun setSkewY(angle: Float)

            companion object {
                val SVG_TRANSFORM_UNKNOWN: Short
                val SVG_TRANSFORM_MATRIX: Short
                val SVG_TRANSFORM_TRANSLATE: Short
                val SVG_TRANSFORM_SCALE: Short
                val SVG_TRANSFORM_ROTATE: Short
                val SVG_TRANSFORM_SKEWX: Short
                val SVG_TRANSFORM_SKEWY: Short
            }
        }

        /**
         * Exposes the JavaScript [SVGTransformList](https://developer.mozilla.org/en/docs/Web/API/SVGTransformList) to Kotlin
         */
        public external abstract class SVGTransformList {
            open val length: Int
            open val numberOfItems: Int

            fun clear()
            fun initialize(newItem: SVGTransform): SVGTransform
            fun insertItemBefore(newItem: SVGTransform, index: Int): SVGTransform
            fun replaceItem(newItem: SVGTransform, index: Int): SVGTransform
            fun removeItem(index: Int): SVGTransform
            fun appendItem(newItem: SVGTransform): SVGTransform
            fun createSVGTransformFromMatrix(matrix: DOMMatrixReadOnly): SVGTransform
            fun consolidate(): SVGTransform?
            fun getItem(index: Int): SVGTransform
        }

        @kotlin.internal.InlineOnly
        public inline operator fun SVGTransformList.get(index: Int): SVGTransform? = asDynamic()[index]

        @kotlin.internal.InlineOnly
        public inline operator fun SVGTransformList.set(index: Int, newItem: SVGTransform) {
            asDynamic()[index] = newItem
        }
    }
}
Exposes the JavaScript

**Exposes the JavaScript**

```kotlin
/**
 * Exposes the JavaScript
 **/
@<https://developer.mozilla.org/en/docs/Web/API/SVGA\nimatedTransformList>

```
/**
 * Exposes the JavaScript [SVGEllipseElement](https://developer.mozilla.org/en/docs/Web/API/SVGEllipseElement) to Kotlin *
 */
public external abstract class SVGEllipseElement : SVGGeometryElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val rx: SVGAnimatedLength
    open val ry: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGLineElement](https://developer.mozilla.org/en/docs/Web/API/SVGLineElement) to Kotlin *
 */
public external abstract class SVGLineElement : SVGGeometryElement {
    open val x1: SVGAnimatedLength
    open val y1: SVGAnimatedLength
    open val x2: SVGAnimatedLength
    open val y2: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGMeshElement](https://developer.mozilla.org/en/docs/Web/API/SVGMeshElement) to Kotlin *
 */
public external abstract class SVGMeshElement : SVGGeometryElement, SVGURIReference {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
    }
}

/**
 * Exposes the JavaScript [SVGAnimatedPoints](https://developer.mozilla.org/en/docs/Web/API/SVGAnimatedPoints) to Kotlin *
 */
public external interface SVGAnimatedPoints {
    val points: SVGPointList
    val animatedPoints: SVGPointList
}

public external abstract class SVGPointList {
    open val length: Int
    open val numberOfItems: Int
    fun clear()
    fun initialize(newItem: DOMPoint): DOMPoint
    fun insertItemBefore(newItem: DOMPoint, index: Int): DOMPoint
    fun replaceItem(newItem: DOMPoint, index: Int): DOMPoint
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2442
DOMPoint
  fun removeItem(index: Int): DOMPoint?
  fun getItem(index: Int): DOMPoint
  fun appendItem(newItem: DOMPoint): DOMPoint

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.get(index: Int): DOMPoint? = asDynamic()[index]

@kotlin.internal.InlineOnly
public inline operator fun SVGPointList.set(index: Int, newItem: DOMPoint) { asDynamic()[index] = newItem }

/**
   * Exposes the JavaScript [SVGPolylineElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolylineElement) to Kotlin
   */
public external abstract class SVGPolylineElement : SVGGeometryElement, SVGAnimatedPoints {

companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
   * Exposes the JavaScript [SVGPolygonElement](https://developer.mozilla.org/en/docs/Web/API/SVGPolygonElement) to Kotlin
   */
public external abstract class SVGPolygonElement : SVGGeometryElement, SVGAnimatedPoints {

companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
   * Exposes the JavaScript [SVGTextContentElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextContentElement) to Kotlin
   */
public external abstract class SVGTextContentElement : SVGGraphicsElement {
    open val textLength: SVGAnimatedLength
    open val lengthAdjust: SVGAnimatedEnumeration
    fun getNumberOfChars(): Int
    fun getComputedTextLength(): Float
    fun getSubStringLength(charnum: Int, nchars: Int): Float
    fun getStartPositionOfChar(charnum: Int): DOMPoint
    fun getEndPositionOfChar(charnum: Int): DOMPoint
    fun getExtentOfChar(charnum: Int): DOMRect
    fun getRotationOfChar(charnum: Int): Float
    fun getCharNumAtPosition(point: DOMPoint): Int
    fun selectSubString(charnum: Int, nchars: Int)

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
   * Exposes the JavaScript [SVGTextPositioningElement](https://developer.mozilla.org/en/docs/Web/API/SVGSVGElement) to Kotlin
   */
public external abstract class SVGTextPositioningElement : SVGTextContentElement {
    open val x: SVGAnimatedLengthList
    open val y: SVGAnimatedLengthList
    open val dx: SVGAnimatedLengthList
    open val dy: SVGAnimatedLengthList

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
open val dy: SVGAnimatedLengthList
    open val rotate: SVGAnimatedNumberList

    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTextElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextElement) to Kotlin
 */
public external abstract class SVGTextElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTSpanElement](https://developer.mozilla.org/en/docs/Web/API/SVGTSpanElement) to Kotlin
 */
public external abstract class SVGTSpanElement : SVGTextPositioningElement {
    companion object {
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGTextPathElement](https://developer.mozilla.org/en/docs/Web/API/SVGTextPathElement) to Kotlin
 */
public external abstract class SVGTextPathElement : SVGTextContentElement, SVGURIReference {
    open val startOffset: SVGAnimatedLength
    open val method: SVGAnimatedEnumeration
    open val spacing: SVGAnimatedEnumeration

    companion object {
        val TEXTPATH_METHODTYPE_UNKNOWN: Short
        val TEXTPATH_METHODTYPE_ALIGN: Short
        val TEXTPATH_METHODTYPE_STRETCH: Short
        val TEXTPATH_SPACINGTYPE_UNKNOWN: Short
        val TEXTPATH_SPACINGTYPE_AUTO: Short
        val TEXTPATH_SPACINGTYPE_EXACT: Short
        val LENGTHADJUST_UNKNOWN: Short
        val LENGTHADJUST_SPACING: Short
        val LENGTHADJUST_SPACINGANDGLYPHS: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
```kotlin
/**
 * Exposes the JavaScript [SVGImageElement](https://developer.mozilla.org/en/docs/Web/API/SVGImageElement) to Kotlin
 */
public external abstract class SVGImageElement : SVGGraphicsElement, SVGURIReference, HTMLOrSVGImageElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength
    open val preserveAspectRatio: SVGAnimatedPreserveAspectRatio
    open var crossOrigin: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript [SVGForeignObjectElement](https://developer.mozilla.org/en/docs/Web/API/SVGForeignObjectElement) to Kotlin
 */
public external abstract class SVGForeignObjectElement : SVGGraphicsElement {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGMarkerElement : SVGElement, SVGFitToViewBox {
    open val refX: SVGAnimatedLength
    open val refY: SVGAnimatedLength
    open val markerUnits: SVGAnimatedEnumeration
    open val markerWidth: SVGAnimatedLength
    open val markerHeight: SVGAnimatedLength
    open val orientType: SVGAnimatedEnumeration
    open val orientAngle: SVGAnimatedAngle
    open var orient: String
    fun setOrientToAuto()
    fun setOrientToAngle(angle: SVGAngle)

    companion object {
        val SVG_MARKERUNITS_UNKNOWN: Short
        val SVG_MARKERUNITS_USERSPACEONUSE: Short
        val SVG_MARKERUNITS_STROKEWIDTH: Short
        val SVG_MARKER_ORIENT_UNKNOWN: Short
        val SVG_MARKER_ORIENT_AUTO: Short
        val SVG_MARKER_ORIENT_ANGLE: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
    }
```

---

**Open Source Used In IOx-FogDirector-AC18 1.17.0 2445**
DOCUMENT_POSITION_DISCONNECTED: Short
val
DOCUMENT_POSITION_FOLLOWING: Short
val
DOCUMENT_POSITION_CONTAINED_BY: Short
val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

* Exposes the JavaScript

[SVGSolidColorElement](https://developer.mozilla.org/en/docs/Web/API/SVGSolidColorElement) to Kotlin

//An public external abstract class SVGSolidColorElement : SVGElement {\n \n companion object {\n \n ELEMENT_NODE: Short \n val
 \n ATTRIBUTE_NODE: Short \n val
 \n TEXT_NODE: Short
 \n CDATA_SECTION_NODE: Short \n val
 \n ENTITY_REFERENCE_NODE: Short \n val
 \n ENTITY_NODE: Short \n val
 \n DOCUMENT_NODE: Short \n val
 \n DOCUMENT_TYPE_NODE: Short \n val
 \n DOCUMENT_FRAGMENT_NODE: Short \n val
 \n DOCUMENT_POSITION_DISCONNECTED: Short \n val
 \n DOCUMENT_POSITION_FOLLOWING: Short
 \n DOCUMENT_POSITION_CONTAINS: Short
 \n DOCUMENT_POSITION_CONTAINED_BY: Short
 \n DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

* Exposes the JavaScript

[SVGGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGGradientElement) to Kotlin

//An public external abstract class SVGGradientElement : SVGElement, SVGURIReference, SVGUnitTypes {\n\n open val gradientUnits: SVGAnimatedEnumeration\n open val gradientTransform: SVGAnimatedTransformList\n\n\n\n \n DOCUMENT_POSITION_DISCONNECTED: Short
val
DOCUMENT_POSITION_FOLLOWING: Short
val
DOCUMENT_POSITION_CONTAINED_BY: Short
val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

* Exposes the JavaScript

[SVGLinearGradientElement](https://developer.mozilla.org/en/docs/Web/API/SVGLinearGradientElement) to Kotlin

//An public external abstract class SVGLinearGradientElement : SVGElement {\n\n open val x1: SVGAnimatedLength\n open val y1: SVGAnimatedLength\n open val x2: SVGAnimatedLength\n open val y2: SVGAnimatedLength\n \n companion object {\n \n ELEMENT_NODE: Short \n val
 \n TEXT_NODE: Short
 \n CDATA_SECTION_NODE: Short \n val
 \n ENTITY_NODE: Short \n val
 \n DOCUMENT_NODE: Short \n val
 \n DOCUMENT_TYPE_NODE: Short \n val
 \n DOCUMENT_FRAGMENT_NODE: Short \n val
 \n DOCUMENT_POSITION_DISCONNECTED: Short
val
DOCUMENT_POSITION_FOLLOWING: Short
val
DOCUMENT_POSITION_CONTAINS: Short
val
DOCUMENT_POSITION_CONTAINED_BY: Short
val
DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

* Exposes the JavaScript
public abstract class SVGRadialGradientElement : SVGGradientElement {
    open val cx: SVGAnimatedLength
    open val cy: SVGAnimatedLength
    open val r: SVGAnimatedLength
    open val fx: SVGAnimatedLength
    open val fy: SVGAnimatedLength
    open val fr: SVGAnimatedLength

    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val SVG_UNIT_TYPE_OBJECTOBJECTBOUNDINGBOX: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public abstract class SVGMeshGradientElement : SVGGradientElement {
    companion object {
        val SVG_SPREADMETHOD_UNKNOWN: Short
        val SVG_SPREADMETHOD_PAD: Short
        val SVG_SPREADMETHOD_REFLECT: Short
        val SVG_SPREADMETHOD_REPEAT: Short
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val SVG_UNIT_TYPE_OBJECTOBJECTBOUNDINGBOX: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public abstract class SVGMeshrowElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public abstract class SVGMeshpatchElement : SVGElement {
    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val DOCUMENT_POSITION_FOLLOWING: Short
val DOCUMENT_POSITION_CONTAINS: Short

companion object {
    val ELEMENT_NODE: Short
    val ATTRIBUTE_NODE: Short
    val TEXT_NODE: Short
    val CDATA_SECTION_NODE: Short
    val ENTITY_REFERENCE_NODE: Short
    val ENTITY_NODE: Short
    val PROCESSING_INSTRUCTION_NODE: Short
    val COMMENT_NODE: Short
    val DOCUMENT_NODE: Short
    val DOCUMENT_TYPE_NODE: Short
    val DOCUMENT_FRAGMENT_NODE: Short
    val NOTATION_NODE: Short
    val DOCUMENT_POSITION_DISCONNECTED: Short
    val DOCUMENT_POSITION_PRECEDING: Short
    val DOCUMENT_POSITION_FOLLOWING: Short
    val DOCUMENT_POSITION_CONTAINS: Short
    val DOCUMENT_POSITION_CONTAINED_BY: Short
    val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
}

/**
 * Exposes the JavaScript
[SVGStopElement](https://developer.mozilla.org/en/docs/Web/API/SVGStopElement) to Kotlin
 */
public external abstract class SVGStopElement : SVGElement {
    open val offset: SVGAnimatedNumber

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/**
 * Exposes the JavaScript
[SVGPatternElement](https://developer.mozilla.org/en/docs/Web/API/SVGPatternElement) to Kotlin
 */
public external abstract class SVGPatternElement : SVGElement, SVGFitToViewBox, SVGURIReference, SVGUnitTypes {
    open val patternUnits: SVGAnimatedEnumeration
    open val patternContentUnits: SVGAnimatedEnumeration
    open val patternTransform: SVGAnimatedTransformList
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength
    open val width: SVGAnimatedLength
    open val height: SVGAnimatedLength

    companion object {
        val SVG_UNIT_TYPE_UNKNOWN: Short
        val SVG_UNIT_TYPE_USERSPACEONUSE: Short
        val SVG_UNIT_TYPE_OBJECTBOUNDINGBOX: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGHatchElement : SVGElement {

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

public external abstract class SVGHatchpathElement : SVGElement {

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}
val DOCUMENT_POSITION_CONTAINED_BY: Short
val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short

*/

/*
 * Exposes the JavaScript [SVGCursorElement](https://developer.mozilla.org/en/docs/Web/API/SVGCursorElement) to Kotlin */

public external abstract class SVGCursorElement : SVGElement, SVGURIReference {
    open val x: SVGAnimatedLength
    open val y: SVGAnimatedLength

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/*
 * Exposes the JavaScript [SVGScriptElement](https://developer.mozilla.org/en/docs/Web/API/SVGScriptElement) to Kotlin */

public external abstract class SVGScriptElement : SVGElement, SVGURIReference, HTMLOrSVGScriptElement {
    open var type: String
    open var crossOrigin: String?

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/*
 * Exposes the JavaScript [SVGAElement](https://developer.mozilla.org/en/docs/Web/API/SVGAElement) to Kotlin */

public external abstract class SVGAElement : SVGGraphicsElement, SVGURIReference {
    open val target: SVGAnimatedString
    open val download: SVGAnimatedString
    open val rel: SVGAnimatedString
    open val relList: SVGAnimatedString
    open val hreflang: SVGAnimatedString
    open val type: SVGAnimatedString

    companion object {
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val COMMENT_NODE: Short
        val DOCUMENT_NODE: Short
        val DOCUMENT_TYPE_NODE: Short
        val DOCUMENT_FRAGMENT_NODE: Short
        val NOTATION_NODE: Short
        val DOCUMENT_POSITION_DISCONNECTED: Short
        val DOCUMENT_POSITION_PRECEDING: Short
        val DOCUMENT_POSITION_FOLLOWING: Short
        val DOCUMENT_POSITION_CONTAINS: Short
        val DOCUMENT_POSITION_CONTAINED_BY: Short
        val DOCUMENT_POSITION_IMPLEMENTATION_SPECIFIC: Short
    }
}

/*
 * Exposes the JavaScript [SVGViewElement](https://developer.mozilla.org/en/docs/Web/API/SVGViewElement) to Kotlin */

public external abstract class SVGViewElement : SVGElement, SVGFitToViewBox, SVGZoomAndPan {
    companion object {
        val SVG_ZOOMANDPAN_UNKNOWN: Short
        val SVG_ZOOMANDPAN_DISABLE: Short
        val SVG_ZOOMANDPAN_MAGNIFY: Short
        val SVG_ZOOMANDPAN_ZOOM: Short
        val SVG_ZOOMANDPAN_RECT: Short
        val SVG_ZOOMANDPAN_Move: Short
        val ELEMENT_NODE: Short
        val ATTRIBUTE_NODE: Short
        val TEXT_NODE: Short
        val CDATA_SECTION_NODE: Short
        val ENTITY_REFERENCE_NODE: Short
        val ENTITY_NODE: Short
        val PROCESSING_INSTRUCTION_NODE: Short
        val DOCUMENT_NODE: Short
    }
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2449
JetBrains s.r.o. and Kotlin Programming Language contributors. Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT! See github.com/kotlin/dukat for details!

package org.w3c.files

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.notifications.*
import org.w3c.workers.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [Blob](https://developer.mozilla.org/en/docs/Web/API/Blob) to Kotlin
 */
public external open class Blob(blobParts: Array<dynamic> = definedExternally, options: BlobPropertyBag = definedExternally) : ImageBitmapSource {
    open val size: Number
    open val type: String
    open val isClosed: Boolean
    fun slice(start: Int = definedExternally, end: Int = definedExternally, contentType: String = definedExternally): Blob
    fun close()
}

public external interface BlobPropertyBag {
    var type: String? /* = "" */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun BlobPropertyBag(type: String? = ""): BlobPropertyBag {
    val o = js("({})")
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript [File](https://developer.mozilla.org/en/docs/Web/API/File) to Kotlin
 */
public external open class File(fileBits: Array<dynamic>, fileName: String, options: FilePropertyBag = definedExternally) : Blob {
    open val name: String
    open val lastModified: Int
}

public external interface FilePropertyBag : BlobPropertyBag {
    var lastModified: Int?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
public inline fun FilePropertyBag(lastModified: Int? = undefined, type: String? = ""): FilePropertyBag {
    val o = js("({})")
    o["lastModified"] = lastModified
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript [FileList](https://developer.mozilla.org/en/docs/Web/API/FileList) to Kotlin
 */
public external abstract class FileList : ItemArrayLike<File> {
    override fun item(index: Int): File?
}

@kotlin.internal.InlineOnly
public operator fun FileList.get(index: Int): File? = asDynamic()[index]

/**
 * Exposes the JavaScript [FileReader](https://developer.mozilla.org/en/docs/Web/API/FileReader) to Kotlin
 */
public external open class FileReader : EventTarget {
    open val readyState: Short
    open val result: dynamic
    open val error: dynamic
    var onloadstart: ((ProgressEvent) -> dynamic)?
    var onprogress: ((ProgressEvent) -> dynamic)?
    var onload: ((Event) -> dynamic)?
    var onabort: ((Event) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    var onloadend: ((Event) -> dynamic)?
    fun readAsArrayBuffer(blob: Blob)
    fun readAsBinaryString(blob: Blob)
    fun readAsText(blob: Blob, label: String = definedExternally)
    fun readAsDataURL(blob: Blob)
    fun abort()

    companion object {
        val EMPTY: Short
        val LOADING: Short
        val DONE: Short
    }
}

@kotlin.internal.InlineOnly
public operator fun FileReader.readAsArrayBuffer(blob: Blob): ArrayBuffer
public operator fun FileReader.readAsBinaryString(blob: Blob): String

/**
 * Exposes the JavaScript [FileReaderSync](https://developer.mozilla.org/en/docs/Web/API/FileReaderSync) to Kotlin
 */
public external open class FileReaderSync {
    fun readAsArrayBuffer(blob: Blob): ArrayBuffer
    fun readAsBinaryString(blob: Blob): String
}

Open Source Used In IOx-FogDirector-AC18 1.17.0 2450
org.w3c.dom.svg.*\nimport org.w3c.dom.url.*\nimport org.w3c.fetch.*\nimport org.w3c.files.*\nimport org.w3c.performance.*\nimport org.w3c.workers.*\n\n* Exposes the JavaScript Notification API to Kotlin

public external interface NotificationOptions {
    var dir: NotificationDirection
    var title: String
    var body: String
    var tag: String
    var image: String
    var icon: String
    var badge: String
    var sound: String
    var vibrate: dynamic
    var requireInteraction: Boolean
    var silent: Boolean
    var noscreen: Boolean
    var renotify: Boolean
    var timestamp: Number
    var data: Any
    var actions: Array<NotificationAction>
}

public external open class Notification(title: String, options: NotificationOptions = definedExternally) : EventTarget {
    open val onclick: ((MouseEvent) -> dynamic)?
    var onerror: ((Event) -> dynamic)?
    open val title: String
    open val dir: NotificationDirection
    open val open val lang: String
    open val open val tag: String
    open val open val image: String
    open val open val icon: String
    open val open val badge: String
    open val open val sound: String
    open val open val vibrate: dynamic
    open val open val requireInteraction: Boolean
    open val open val silent: Boolean
    open val open val noscreen: Boolean
    open val open val renotify: Boolean
    open val open val timestamp: Number
    open val open val data: Any
    open val open val actions: Array<NotificationAction>
    open val open val undefined
    open val open val null
}

public external open class NotificationPermission(userID: String?, timeout: Number?) { // return NotificationPermission
    open fun requestPermission(): Promise<NotificationPermission>
}

public external open class NotificationPermission { // val permission: NotificationPermission
    open val maxActions: Int
    open val actions: Array<NotificationAction>
    fun close()

    companion object {
        val maxActions: Int
        fun actions
    }

    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun requestPermission(deprecatedCallback: NotificationPermission -> Unit = definedExternally): Promise<NotificationPermission>?

    public external companion object {
        public val undefined
    }

    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: dynamic
    open val requireInteraction: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val renotify: Boolean
    open val timestamp: Number
    open val data: Any
    open val actions: Array<NotificationAction>
}

public external open class NotificationOptions { // void onerror
    open val value: Int
    open val maxActions: Int
    open val actions: Array<NotificationAction>
    fun requestPermission(): Promise<NotificationPermission>

    companion object {
        val undefined
    }

    open val noscreen: Boolean
    open val requireInteraction: Boolean
    open val sticky: Boolean
    open val data: Any
    open val actions: Array<out NotificationAction>
    fun requestPermission(deprecatedCallback: NotificationPermission -> Unit = definedExternally): Promise<NotificationPermission>?

    public external companion object {
        public val undefined
    }

    open val title: String
    open val dir: NotificationDirection
    open val lang: String
    open val body: String
    open val tag: String
    open val image: String
    open val icon: String
    open val badge: String
    open val sound: String
    open val vibrate: dynamic
    open val requireInteraction: Boolean
    open val silent: Boolean
    open val noscreen: Boolean
    open val renotify: Boolean
    open val timestamp: Number
    open val data: Any
    open val actions: Array<NotificationAction>
}
/**
 * Exposes the JavaScript [ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin
 */
public external abstract class ServiceWorker : EventTarget, AbstractWorker, UnionMessagePortOrServiceWorker,
    UnionClientOrMessagePortOrServiceWorker {
    open val scriptURL: String
    open val state: ServiceWorkerState
    open var onstatechange: ((Event) -> dynamic)?
    fun postMessage(message: Any?,
        transfer: Array<dynamic> = definedExternally)
}

/**
 * Exposes the JavaScript [ServiceWorkerRegistration](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerRegistration) to Kotlin
 */
public external abstract class ServiceWorkerRegistration : EventTarget {
    open val installing: ServiceWorker?
    open val waiting: ServiceWorker?
    open val active: ServiceWorker?
    open val scope: String
    open var onupdatefound: ((Event) -> dynamic)?
    open val APISpace: dynamic
    fun update(): Promise<Unit>
    fun unregister(): Promise<Boolean>
    fun showNotification(title: String, options: NotificationOptions = definedExternally): Promise<Unit>
    fun methodName(): Promise<dynamic>
}

/**
 * Exposes the JavaScript [ServiceWorkerContainer](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerContainer) to Kotlin
 */
public external

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

// NOTE: THIS FILE IS AUTO-GENERATED, DO NOT EDIT!
// See github.com/kotlin/dukat for details

package org.w3c.workers

import kotlin.js.*
import org.khronos.webgl.*
import org.w3c.css.masking.*
import org.w3c.dom.*
import org.w3c.dom.clipboard.*
import org.w3c.dom.css.*
import org.w3c.dom.events.*
import org.w3c.dom.mediacapture.*
import org.w3c.dom.parsing.*
import org.w3c.dom.pointerevents.*
import org.w3c.dom.svg.*
import org.w3c.dom.url.*
import org.w3c.fetch.*
import org.w3c.files.*
import org.w3c.notifications.*
import org.w3c.performance.*
import org.w3c.xhr.*

/**
 * Exposes the JavaScript [ServiceWorker](https://developer.mozilla.org/en/docs/Web/API/ServiceWorker) to Kotlin
 */
public external

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
*/
public external abstract class ServiceWorkerContainer : EventTarget {
    open val ready: Promise<ServiceWorkerRegistration>
    open var oncontrollerchange: ((Event) -> dynamic)?
    fun register(scriptURL: String, options: RegistrationOptions = definedExternally): Promise<ServiceWorkerRegistration>
    fun getRegistration(clientURL: String = definedExternally): PromiseAny?
    fun getRegistrations(): PromiseArray<ServiceWorkerRegistration>?
}

public external interface RegistrationOptions {
    var scope: String?
    var type: WorkerType? /* = WorkerType.CLASSIC */
}

@kotlin.internal.InlineOnly
public inline fun RegistrationOptions(scope: String? = undefined, type: WorkerType? = WorkerType.CLASSIC): RegistrationOptions {
    val o = js("({})")
    o["scope"] = scope
    o["type"] = type
    return o
}

/**
 * Exposes the JavaScript [ServiceWorkerMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerMessageEvent) to Kotlin
 */
public external open class ServiceWorkerMessageEvent(type: String, eventInitDict: ServiceWorkerMessageEventInit = definedExternally) : Event {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val source: UnionMessagePortOrServiceWorker?
    open val ports: Array<MessagePort>?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }

    public external interface ServiceWorkerMessageEventInit : EventInit {
        var data: Any?
            get() = definedExternally
            set(value) = definedExternally
        var origin: String?
            get() = definedExternally
            set(value) = definedExternally
        var lastEventId: String?
            get() = definedExternally
            set(value) = definedExternally
        var source: UnionMessagePortOrServiceWorker?
            get() = definedExternally
            set(value) = definedExternally
        var ports: Array<MessagePort>?
            get() = definedExternally
            set(value) = definedExternally
    }

    @kotlin.internal.InlineOnly
        val o = js("({})")
        o["data"] = data
        o["origin"] = origin
        o["lastEventId"] = lastEventId
        o["source"] = source
        o["ports"] = ports
        o["bubbles"] = bubbles
        o["cancelable"] = cancelable
        o["composed"] = composed
        return o
    }

    /**
     * Exposes the JavaScript [ServiceWorkerGlobalScope](https://developer.mozilla.org/en/docs/Web/API/ServiceWorkerGlobalScope) to Kotlin
     */
    public external abstract class ServiceWorkerGlobalScope : WorkerGlobalScope {
        open val clients: Clients
        open val registration: ServiceWorkerRegistration
        open var oninstall: ((Event) -> dynamic)?
        open var onactivate: ((Event) -> dynamic)?
        open var onfetch: ((FetchEvent) -> dynamic)?
        open var onforeignfetch: ((Event) -> dynamic)?
        open var onmessage: ((MessageEvent) -> dynamic)?
        open var onnotificationclick: ((NotificationEvent) -> dynamic)?
        open var onfunctionalevent: ((Event) -> dynamic)?
        fun skipWaiting(): Promise<Unit>
    }

    /**
     * Exposes the JavaScript [Client](https://developer.mozilla.org/en/docs/Web/API/Client) to Kotlin
     */
    public external abstract class Client : UnionClientOrMessagePortOrServiceWorker {
        open val url: String
        open val frameType: FrameType
        open val id: String
        fun postMessage(message: Any?, transfer: Array<dynamic> = definedExternally)
    }

    /**
     * Exposes the JavaScript [WindowClient](https://developer.mozilla.org/en/docs/Web/API/WindowClient) to Kotlin
     */
    public external abstract class WindowClient : Client {
        open val visibilityState: dynamic
        open val focused: Boolean
        fun focus(): Promise<WindowClient>
        fun navigate(url: String): Promise<WindowClient>
    }

    /**
     * Exposes the JavaScript [Clients](https://developer.mozilla.org/en/docs/Web/API/Clients) to Kotlin
     */
    public external abstract class Clients {
        fun get(id: String): Promise<WindowClient>
        fun getAll(options: ClientQueryOptions = definedExternally): PromiseArray<WindowClient>?
        fun openWindow(url: String): Promise<WindowClient>
        fun claim(): Promise<Unit>
    }

    public external interface ClientQueryOptions {
        var includeUncontrolled: Boolean
    }

    @kotlin.internal.InlineOnly
    public inline fun Clients(open val includeUncontrolled: Boolean?
)}
Open Source Used In IOx-FogDirector-AC18 1.17.0

Boolean? /* = false */

get() = definedExternally

set(value) = definedExternally

var type:

ClientType? /* = ClientType.WINDOW */

get() = definedExternally

set(value) = definedExternally

var type:

ClientType? /* = ClientType.WINDOW */

ClientQueryOptions {\n  val o = js("{\"\"})\n  o[\"includeUncontrolled\"] = includeUncontrolled\n  o[\"type\"] = type\n  return o\n}\n
/* Exposes the JavaScript [ExtendableEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableEvent) to Kotlin

*\n
/public external open class ExtendableEvent(type: String, eventInitDict: ExtendableEventInit = definedExternally) : Event {

fun waitFor(f: Promise<Any?>)?\n
  companion object {
    val NONE: Short
    val CAPTURING_PHASE: Short
    val AT_TARGET: Short
    val BUBBLING_PHASE: Short
  }

  fun registerForeignFetch(options: ForeignFetchOptions?)\n
    companion object {
      val NONE: Short
      val CAPTURING_PHASE: Short
      val AT_TARGET: Short
      val BUBBLING_PHASE: Short
    }

    fun respondWith(r: Promise<Response>)

    fun respondWith(r: Promise<Response>)

    fun respondWith(r: Promise<Response>)

    fun respondWith(r: Promise<Response>)

}
    val o = js("({})")
    o["response"] = response
    o["origin"] = origin
    o["headers"] = headers
    return o
}

/**
 * Exposes the JavaScript
 * [ExtendableMessageEvent](https://developer.mozilla.org/en/docs/Web/API/ExtendableMessageEvent) to Kotlin
 */
public external open class ExtendableMessageEvent(type: String, eventInitDict: ExtendableMessageEventInit = definedExternally) : ExtendableEvent {
    open val data: Any?
    open val origin: String
    open val lastEventId: String
    open val source: UnionClientOrMessagePortOrServiceWorker?
    open val ports: Array<MessagePort>?

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ExtendableMessageEventInit : ExtendableEventInit {
    var data: Any?
        get() = definedExternally
        set(value) = definedExternally
    var origin: String?
        get() = definedExternally
        set(value) = definedExternally
    var lastEventId: String?
        get() = definedExternally
        set(value) = definedExternally
    var source: UnionClientOrMessagePortOrServiceWorker?
        get() = definedExternally
        set(value) = definedExternally
    var ports: Array<MessagePort>?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["data"] = data
    o["origin"] = origin
    o["lastEventId"] = lastEventId
    o["source"] = source
    o["ports"] = ports
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/**
 * Exposes the JavaScript
 * [Cache](https://developer.mozilla.org/en/docs/Web/API/Cache) to Kotlin
 */
public external abstract class Cache {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun matchAll(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally): Promise<Array<Response>>
    fun add(request: dynamic): Promise<Unit>
    fun addAll(requests: Array<dynamic>): Promise<Unit>
    fun put(request: dynamic, response: Response): Promise<Unit>
    fun delete(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Boolean>
    fun keys(request: dynamic = definedExternally, options: CacheQueryOptions = definedExternally): Promise<Array<Request>>
}

public external interface CacheQueryOptions {
    var ignoreSearch: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ignoreMethod: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var ignoreVary: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var cacheName: String?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["ignoreSearch"] = ignoreSearch
    o["ignoreMethod"] = ignoreMethod
    o["ignoreVary"] = ignoreVary
    o["cacheName"] = cacheName
    return o
}

public external interface CacheBatchOperation {
    var type: String?
        get() = definedExternally
        set(value) = definedExternally
    var request: Request?
        get() = definedExternally
        set(value) = definedExternally
    var response: Response?
        get() = definedExternally
        set(value) = definedExternally
    var options: CacheQueryOptions?
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["type"] = type
    o["request"] = request
    o["response"] = response
    o["options"] = options
    return o
}

/**
 * Exposes the JavaScript
 * [CacheStorage](https://developer.mozilla.org/en/docs/Web/API/CacheStorage) to Kotlin
 */
public external abstract class CacheStorage {
    fun match(request: dynamic, options: CacheQueryOptions = definedExternally): Promise<Any?>
    fun has(cacheName: String): Promise<Boolean>
    fun open(cacheName: String): Promise<Cache>
    fun delete(cacheName: String): Promise<Boolean>
    fun keys():

Open Source Used In IOX-FogDirector-AC18 1.17.0 2455
Promise<Array<String>>
}

public external open class FunctionalEvent : ExtendableEvent {
    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface UnionMessagePortOrServiceWorker

public external interface UnionClientOrMessagePortOrServiceWorker

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ServiceWorkerState {
    companion object
}

public inline val ServiceWorkerState.Companion.INSTALLING: ServiceWorkerState get() = "installing".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.INSTALLED: ServiceWorkerState get() = "installed".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATING: ServiceWorkerState get() = "activating".asDynamic().unsafeCast<ServiceWorkerState>()

public inline val ServiceWorkerState.Companion.ACTIVATED: ServiceWorkerState get() = "activated".asDynamic().unsafeCast<ServiceWorkerState>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface FrameType {
    companion object
}

public inline val FrameType.Companion.AUXILIARY: FrameType get() = "auxiliary".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.TOP_LEVEL: FrameType get() = "top-level".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NESTED: FrameType get() = "nested".asDynamic().unsafeCast<FrameType>()

public inline val FrameType.Companion.NONE: FrameType get() = "none".asDynamic().unsafeCast<FrameType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface ClientType {
    companion object
}

public inline val ClientType.Companion.WINDOW: ClientType get() = "window".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.WORKER: ClientType get() = "worker".asDynamic().unsafeCast<ClientType>()

public inline val ClientType.Companion.SHAREDWORKER: ClientType get() = "sharedworker".asDynamic().unsafeCast<ClientType>()

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external class XMLHttpRequestEventTarget : EventTarget {
    open var onloadstart: ((ProgressEvent) -> dynamic)?
    open var onprogress: ((ProgressEvent) -> dynamic)?
    open var onabort: ((Event) -> dynamic)?
    open var onerror: ((Event) -> dynamic)?
    open var onload: ((Event) -> dynamic)?
    open var ontimeout: ((Event) -> dynamic)?
    open var onloadend: ((Event) -> dynamic)?
}

public abstract class XMLHttpRequestUpload : XMLHttpRequestEventTarget

public abstract class XMLHttpRequest : XMLHttpRequestEventTarget {
    var onreadystatechange: ((Event) -> dynamic)?
    open val readyState: Short
    var timeout: Int
    var withCredentials: Boolean
    open val upload: XMLHttpRequestUpload
    open val responseURL: String
    open val status: Short
    open val statusText:
String
    var responseType: XMLHttpRequestResponseType
    open val response: Any?
    open val responseText: String
    open val responseXML: Document?
    fun setRequestHeader(name: String, value: String)
    fun send(body: dynamic = definedExternally)
    fun abort()
    fun getResponseHeader(name: String): String?
    fun getAllResponseHeaders(): String
    fun overrideMimeType(mime: String)

    companion object {
        val UNSENT: Short
        val OPENED: Short
        val HEADERS_RECEIVED: Short
        val LOADING: Short
        val DONE: Short
    }
}

/**
 * Exposes the JavaScript
 [FormData](https://developer.mozilla.org/en/docs/Web/API/FormData) to Kotlin
 */
public external open class FormData(form: HTMLFormElement = definedExternally) {
    fun append(name: String, value: String)
    fun append(name: String, value: Blob, filename: String = definedExternally)
    fun delete(name: String)
    fun get(name: String): dynamic
    fun getAll(name: String): Array<dynamic>
    fun has(name: String): Boolean
    fun set(name: String, value: String)
    fun set(name: String, value: Blob, filename: String = definedExternally)
}

/**
 * Exposes the JavaScript
 [ProgressEvent](https://developer.mozilla.org/en/docs/Web/API/ProgressEvent) to Kotlin
 */
public external open class ProgressEvent(type: String, eventInitDict: ProgressEventInit = definedExternally) : Event {
    open val lengthComputable: Boolean
    open val loaded: Number
    open val total: Number

    companion object {
        val NONE: Short
        val CAPTURING_PHASE: Short
        val AT_TARGET: Short
        val BUBBLING_PHASE: Short
    }
}

public external interface ProgressEventInit : EventInit {
    var lengthComputable: Boolean? /* = false */
        get() = definedExternally
        set(value) = definedExternally
    var loaded: Number? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
    var total: Number? /* = 0 */
        get() = definedExternally
        set(value) = definedExternally
}

@kotlin.internal.InlineOnly
    val o = js("({})")
    o["lengthComputable"] = lengthComputable
    o["loaded"] = loaded
    o["total"] = total
    o["bubbles"] = bubbles
    o["cancelable"] = cancelable
    o["composed"] = composed
    return o
}

/* please, don't implement this interface! */
@Suppress("NESTED_CLASS_IN_EXTERNAL_INTERFACE")
public external interface XMLHttpRequestResponseType {
    companion object
}

public inline val XMLHttpRequestResponseType.Companion.EMPTY: XMLHttpRequestResponseType get() = "".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.ARRAYBUFFER: XMLHttpRequestResponseType get() = "arraybuffer".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.BLOB: XMLHttpRequestResponseType get() = "blob".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.DOCUMENT: XMLHttpRequestResponseType get() = "document".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.JSON: XMLHttpRequestResponseType get() = "json".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

public inline val XMLHttpRequestResponseType.Companion.TEXT: XMLHttpRequestResponseType get() = "text".asDynamic().unsafeCast<XMLHttpRequestResponseType>()

/* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors. */

/* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */
using [UseExperimental].
* or by being annotated with that marker themselves, effectively causing further
propagation of that experimental aspect.

This class is deprecated in favor of a more general approach
provided by [RequiresOptIn]/[OptIn].

*\@Target(ANNOTATION_CLASS)\@Retention(BINARY)\@SinceKotlin("1.2")\@RequireKotlin("1.2.50 \ versionKind = RequireKotlinVersionKind.COMpiler_VERSION)\@Deprecated("Please use RequiresOptIn instead.")\)un

\public annotation class Experimental(val level: Level = Level.ERROR) { \n
*\@Retention(AnnotationRetention.BINARY)\@SinceKotlin("1.3")\@RequiresOptIn(level = Level.ERROR)\@Target(\CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE, TYPEALIAS)\@MustBeDocumented\@OptIn(ExperimentalStdlibApi::class)\@Suppress("DEPRECATION")\@Experimental\}

\/**
 * This annotation marks the standard library API that is considered experimental and is not subject to the
 * general compatibility guarantees.[https://kotlinlang.org/docs/reference/evolution/components-stability.html] given for the
 * standard library.
 *
 * Beware using the annotated API especially if you're developing a library, since your library
 * might become binary incompatible with the future versions of the standard library.
 *
 * Any usage of a declaration annotated with `@ExperimentalStdlibApi` must be accepted either by
 * annotating that usage with the
 * [OptIn] annotation, e.g. `@OptIn(ExperimentalStdlibApi::class)` or by using the compiler argument `-Xopt-
 * in=kotlin.ExperimentalStdlibApi`
 *
 * This extension marks the standard library API that is considered experimental and is not subject to the
 * general compatibility guarantees.[https://kotlinlang.org/docs/reference/evolution/components-stability.html] given for the
 * standard library.
 *
 * Beware using the annotated API especially if you're developing a library, since your library
 * might become binary incompatible with the future versions of the standard library.
 *
 * Any usage of a declaration annotated with `@ExperimentalStdlibApi` must be accepted either by
 * annotating that usage with the
 * [OptIn] annotation, e.g. `@OptIn(ExperimentalStdlibApi::class)` or by using the compiler argument `-Xopt-
 * in=kotlin.ExperimentalStdlibApi`
 *
this annotation is experimental, see [ExperimentalTypeInference] on how to opt-in for it.

@Target(VALUE_PARAMETER, FUNCTION, PROPERTY)
@Retention(AnnotationRetention.BINARY)
@SinceKotlin("1.3")
@ExperimentalTypeInference

The experimental multiplatform support API marker. @n
Any usage of a declaration annotated with `@ExperimentalMultiplatform` must be accepted either by
appending the compiler argument `-Xopt-in=kotlin.ExperimentalMultiplatform`
with the [OptIn] annotation, e.g. `@OptIn(ExperimentalMultiplatform::class)`.

* @n

This annotation is only applicable to `expect` annotation classes in multi-platform projects and marks that class as
"optional". @n
Optional expected class is allowed to have no corresponding actual class on the platform. Optional
annotations can only be used to annotate something, not as types in signatures. If an optional annotation has no

The default message is: `"This declaration is experimental and its usage should be marked with 'Marker'
or '@OptIn(Marker::class)'", where 'Marker' is the opt-in requirement marker.`
Open Source Used In IOx-FogDirector-AC18 1.17.0 2460

[OptIn], its usages are **not** required to opt in to that API.

* This class requires opt-in itself and can only be used with the compiler argument -Xopt-in=kotlin.RequiresOptIn.

* @Target(CLASS, PROPERTY, LOCAL_VARIABLE, VALUE_PARAMETER, CONSTRUCTOR, FUNCTION, PROPERTY_GETTER, PROPERTY_SETTER, EXPRESSION, FILE,

* @Retention(SOURCE) @SinceKotlin("1.3") @RequireKotlin("1.3.70")

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* Provides a skeletal implementation of the read-only [Collection] interface.

* @param E the type of elements contained in the collection.

* The collection is covariant on its element type.

* @SinceKotlin("1.1")

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [List] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* @package kotlin.collections

* @package kotlin.js

* @SinceKotlin("1.1")

* Provides a skeletal implementation of the read-only [Map] interface.

* Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
Programming Language contributors.* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.*

*native.concurrent.SharedImmutable.*

```kotlin
open class ArrayDeque<E> {
    private var head: Int = 0
    private var elementData: Array<Any?>

    override var size: Int = 0
        private set

    /**
     * Constructs an empty deque.
     */
    public constructor() {
        elementData = emptyElementData
    }

    /**
     * Constructs a deque that contains the same elements as the specified elements in the same order.
     */
    public constructor(elements: Collection<E>) {
        elementData = elements.toTypedArray()
        size = elementData.size
        if (elementData.isEmpty())
            elementData = emptyElementData
    }

    /**
     * Ensures that the capacity of this deque is at least equal to the specified [capacity].
     * If the current capacity is less than the [capacity], a new backing storage is allocated with greater capacity.
     * Otherwise, this method takes no action and simply returns.
     */
    private fun ensureCapacity(capacity: Int) {
        if (capacity < 0) throw IllegalStateException("Deque is too big.")
        if (capacity <= elementData.size)
            return
        if (elementData === emptyElementData) {
            val newCapacity = newCapacity(elementData.size, capacity)
            copyElements(newCapacity)
        }

        // made internal for testing
        val newCapacity = newCapacity(capacity, capacity)
        copyElements(newCapacity)
    }

    // made internal for testing
    private fun newCapacity(oldCapacity: Int, capacity: Int): Int {
        var newCapacity = oldCapacity + (oldCapacity shr 1)
        if (newCapacity - capacity < 0)
            newCapacity = capacity
        if (newCapacity - maxArraySize > 0)
            newCapacity = if (capacity > maxArraySize) Int.MAX_VALUE else maxArraySize
        return newCapacity
    }

    /**
     * Creates a new array with the specified [newCapacity] size and copies elements in the [elementData] array to it.
     */
    private fun copyElements(newCapacity: Int) {
        val newElements = arrayOfNulls<Any?>(newCapacity)
        elementData.copyInto(newElements, 0, head, elementData.size)
        elementData.copyInto(newElements, elementData.size - head, 0, head)
        head = 0
        elementData = newElements
    }

    @kotlin.internal.InlineOnly
    private inline fun internalGet(internalIndex: Int): E {
        return elementData[internalIndex] as E
    }

    private fun positiveMod(index: Int): Int = if (index >= elementData.size) index - elementData.size else index

    private fun negativeMod(index: Int): Int = if (index < 0) index + elementData.size else index

    @kotlin.internal.InlineOnly
    private inline fun internalIndex(index: Int): Int = positiveMod(head + index)

    private fun incremented(index: Int): Int = if (index == elementData.lastIndex) 0 else index + 1

    private fun decremented(index: Int): Int = if (index == 0) elementData.lastIndex else index - 1

    override fun isEmpty(): Boolean = size == 0

    *\n    * Returns the first element, or throws [NoSuchElementException] if this deque is empty.
    */
    public fun addFirst(element: E) {
        ensureCapacity(size + 1)
        head = decremented(head)
        elementData.copyInto(newElements, head, 0, elementData.size)
elementData[head] = element

size += 1

/**
 * Appends the specified [element] to this deque.
 */
public fun addLast(element: E) {
    ensureCapacity(size + 1)
    elementData[internalIndex(size)] = element
    size += 1
}

/**
 * Removes the first element from this deque and returns that removed element, or throws [NoSuchElementException] if this deque is empty.
 */
public fun removeFirst(): E {
    if (isEmpty()) throw NoSuchElementException("ArrayDeque is empty.")

    val element = internalGet(head)

    elementData[head] = null
    head = incremented(head)
    size -= 1

    return element
}

/**
 * Removes the first element from this deque and returns that removed element, or throws `null` if this deque is empty.
 */
public fun removeFirstOrNull(): E? = if (isEmpty()) null else removeFirst()

/**
 * Removes the last element from this deque and returns that removed element, or throws `null` if this deque is empty.
 */
public fun removeLast(): E {
    if (isEmpty()) throw NoSuchElementException("ArrayDeque is empty.")

    val internalLastIndex = internalIndex(lastIndex)
    val element = internalGet(internalLastIndex)
    elementData[internalLastIndex] = null

    size -= 1

    return element
}

/**
 * Removes the last element from this deque and returns `null` if this deque is empty.
 */
public fun removeLastOrNull(): E? = if (isEmpty()) null else removeLast()
}\n\n public override fun addAll(elements: Collection<E>): Boolean {\n
if (elements.isEmpty()) return false\n
ensureCapacity(this.size + elements.size)\n
copyCollectionElements(internalIndex(size), elements)\n
return true\n }\n\n public override fun addAll(index: Int, elements: Collection<E>): Boolean {\n
AbstractList.checkPositionIndex(index, size)\n\n
if (elements.isEmpty()) {\n
return false\n
} else if
(index == size) {\n
return addAll(elements)\n
}\n\n
ensureCapacity(this.size + elements.size)\n\n
val tail = internalIndex(size)\n
val internalIndex = internalIndex(index)\n
val elementsSize =
elements.size\n\n
if (index < (size + 1) shr 1) {\n
// closer to the first element -> shift preceding
elements\n\n
var shiftedHead = head - elementsSize\n\n
if (internalIndex >= head) {\n
if
(shiftedHead >= 0) {\n
elementData.copyInto(elementData, shiftedHead, head, internalIndex)\n
} else { // head < tail, insertion leads to head >= tail\n
shiftedHead += elementData.size\n
val
elementsToShift = internalIndex - head\n
val shiftToBack = elementData.size - shiftedHead\n\n
if (shiftToBack >= elementsToShift) {\n
elementData.copyInto(elementData, shiftedHead, head,
internalIndex)\n
} else {\n
elementData.copyInto(elementData, shiftedHead, head, head +
shiftToBack)\n
elementData.copyInto(elementData, 0, head + shiftToBack, internalIndex)\n
}\n
}\n
} else { // head > tail, internalIndex < tail\n
elementData.copyInto(elementData,
shiftedHead, head, elementData.size)\n
if (elementsSize >= internalIndex) {\n
elementData.copyInto(elementData, elementData.size - elementsSize, 0, internalIndex)\n
} else {\n
elementData.copyInto(elementData, elementData.size - elementsSize, 0, elementsSize)\n
elementData.copyInto(elementData, 0, elementsSize, internalIndex)\n
}\n
}\n
head =
shiftedHead\n
copyCollectionElements(negativeMod(internalIndex - elementsSize), elements)\n
} else
{\n
// closer to the last element -> shift succeeding elements\n\n
val shiftedInternalIndex =
internalIndex + elementsSize\n\n
if (internalIndex < tail) {\n
if (tail + elementsSize <=
elementData.size) {\n
elementData.copyInto(elementData, shiftedInternalIndex, internalIndex, tail)\n
} else { // head < tail, insertion leads to head >= tail\n
if (shiftedInternalIndex >= elementData.size)
{\n
elementData.copyInto(elementData, shiftedInternalIndex - elementData.size, internalIndex, tail)\n
} else {\n
val shiftToFront = tail + elementsSize - elementData.size\n
elementData.copyInto(elementData, 0, tail - shiftToFront, tail)\n
elementData.copyInto(elementData,
shiftedInternalIndex, internalIndex, tail - shiftToFront)\n
}\n
}\n
} else { // head > tail,
internalIndex > head\n
elementData.copyInto(elementData, elementsSize, 0, tail)\n
if
(shiftedInternalIndex >= elementData.size) {\n
elementData.copyInto(elementData, shiftedInternalIndex
- elementData.size, internalIndex, elementData.size)\n
} else {\n
elementData.copyInto(elementData, 0, elementData.size - elementsSize, elementData.size)\n
elementData.copyInto(elementData, shiftedInternalIndex, internalIndex, elementData.size - elementsSize)\n
}\n
}\n
copyCollectionElements(internalIndex, elements)\n
}\n\n
return true\n }\n\n public
override fun get(index: Int): E {\n
AbstractList.checkElementIndex(index, size)\n\n
return
internalGet(internalIndex(index))\n }\n\n public override fun set(index: Int, element: E): E {\n
AbstractList.checkElementIndex(index, size)\n\n
val internalIndex = internalIndex(index)\n
val oldElement
= internalGet(internalIndex)\n
elementData[internalIndex] = element\n\n
return oldElement\n }\n\n
public override fun contains(element: E): Boolean = indexOf(element) != -1\n\n public override fun
indexOf(element: E): Int {\n
val tail = internalIndex(size)\n\n
if (head < tail) {\n
for (index in head
until tail) {\n
if (element == elementData[index]) return index - head\n
}\n
} else if (head >=
tail) {\n
for (index in head until elementData.size) {\n
if (element == elementData[index]) return
index - head\n
}\n
for (index in 0 until tail) {\n
if (element == elementData[index]) return
index + elementData.size - head\n
}\n
}\n\n
return -1\n }\n\n public override fun
lastIndexOf(element: E): Int {\n
val tail = internalIndex(size)\n\n
if (head < tail) {\n
for (index in tail
- 1 downTo head) {\n
if (element == elementData[index]) return index - head\n
}\n
} else if
(head > tail) {\n
for (index in tail - 1 downTo 0) {\n
if (element == elementData[index]) return
index + elementData.size - head\n
}\n
for (index in elementData.lastIndex downTo head) {\n

Open Source Used In IOx-FogDirector-AC18 1.17.0 2465


the first values of each pair from this array.

*second* list is built from the second values of each pair from this array.

*sample* collections.Arrays.Transformations.unzipArray

/**
 * Returns true if this nullable array is either null or empty.
 * @sample samples.collections.Arrays.usage
 */
@SinceKotlin("1.3")
@PublishedApi
fun <T> T?.isNullOrEmpty(): Boolean {
    return this == null || this.isEmpty()
}

/**
 * Returns this array if it's not empty or the result of calling [defaultValue] function if
 * the array is empty.
 * @sample samples.collections.Arrays.usage
 */
@SinceKotlin("1.3")
@PublishedApi
fun <T, R> T?.ifEmpty(defaultValue: () -> R): R where T : Array<*>, T : R =
    if (isEmpty()) defaultValue() else this

@UseExperimental(ExperimentalUnsignedTypes::class)
@PublishedApi
private fun <T> T?.ifEmpty(result: StringBuilder, processed: MutableList<T>) {
    if (this in processed) {
        result.append("[...]")
        return
    }
    processed.add(this)
    result.append('[

    for (i in indices) {
        if (i != 0) {
            result.append(', ')
        }
        val element = this[i]
        when (element) {
            null -> result.append("null")
            is Array<*> -> element.ifEmpty(result, processed)
            is ByteArray -> result.append(element.contentToString())
            is ShortArray -> result.append(element.contentToString())
            is IntArray -> result.append(element.contentToString())
            is LongArray -> result.append(element.contentToString())
            is FloatArray -> result.append(element.contentToString())
            is DoubleArray -> result.append(element.contentToString())
            is CharArray -> result.append(element.contentToString())
            is BooleanArray -> result.append(element.contentToString())
            is UByteArray -> result.append(element.contentToString())
            is UShortArray -> result.append(element.contentToString())
            is UIntArray -> result.append(element.contentToString())
            is ULongArray -> result.append(element.contentToString())
            else ->
        }
    }
    return result.toString()
}

@UseExperimental(ExperimentalUnsignedTypes::class)
private fun <T> T?.contentDeepToString(processed: MutableList<T>): String {
    if (this in processed) {
        return "[...]
    }
    processed.add(this)
    return buildString {
        contentDeepToStringInternal(this, mutableListOf())
    }
}

private fun <T> T?.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<T>) {
    if (this in processed) {
        result.append("[...")
        return
    }
    processed.add(this)
    result.append('[

    for (i in indices) {
        if (i != 0) {
            result.append(', ')
        }
        val element = this[i]
        when (element) {
            null -> result.append("null")
            is Array<*> ->
                element.contentDeepToStringInternal(result, processed)
            is ByteArray ->
                result.append(element.contentToString())
            is ShortArray ->
                result.append(element.contentToString())
            is IntArray ->
                result.append(element.contentToString())
            is LongArray ->
                result.append(element.contentToString())
            is FloatArray ->
                result.append(element.contentToString())
            is DoubleArray ->
                result.append(element.contentToString())
            is CharArray ->
                result.append(element.contentToString())
            is BooleanArray ->
                result.append(element.contentToString())
            is UByteArray ->
                result.append(element.contentToString())
            is UShortArray ->
                result.append(element.contentToString())
            is UIntArray ->
                result.append(element.contentToString())
            is ULongArray ->
                result.append(element.contentToString())
            else ->
        }
    }
    return result.append(']')
}

fun <T, R> Array<out T>.unzip(): Pair<List<T>, List<R>> {
    val listT = ArrayList<T>(size)
    val listR = ArrayList<R>(size)
    for (pair in this) {
        listT.add(pair.first)
        listR.add(pair.second)
    }
    return listT to listR
}

/**
 * Returns `true` if this nullable array is either null or empty.
 * @sample samples.collections.Arrays.collectionUsage
 */
@SinceKotlin("1.3")
@PublishedApi
@kotlin.internal.InlineOnly
fun <T> T?.isNullOrEmpty(): Boolean {
    return this == null || this.isEmpty()
}

/**
 * Returns this array if it's not empty or the result of calling [defaultValue] function if
 * the array is empty.
 * @sample samples.collections.Arrays.usage
 */
@SinceKotlin("1.3")
@PublishedApi
@kotlin.internal.InlineOnly
fun <T, R> T?.ifEmpty(defaultValue: () -> R): R where T : Array<*>, T : R =
    if (isEmpty()) defaultValue() else this

@UseExperimental(ExperimentalUnsignedTypes::class)
@PublishedApi
@kotlin.jvm.JvmName("contentDeepToString")
@kotlin.js.JsName("contentDeepToStringImpl")
internal fun <T> Array<out T>.contentDeepToStringImpl(): String {
    if (this === other) return true
    if (this.size != other.size) return false

    for (i in indices) {
        val v1 = this[i]
        val v2 = other[i]

        if (v1 === v2) {
            continue
        } else if (v1 == null || v2 == null) {
            return false
        }

        when {
            v1 is Array<*> && v2 is Array<*> -> if (!v1.contentDeepEquals(v2)) return false
            v1 is ByteArray && v2 is ByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is ShortArray && v2 is ShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is IntArray && v2 is IntArray -> if (!v1.contentEquals(v2)) return false
            v1 is LongArray && v2 is LongArray -> if (!v1.contentEquals(v2)) return false
            v1 is FloatArray && v2 is FloatArray -> if (!v1.contentEquals(v2)) return false
            v1 is DoubleArray && v2 is DoubleArray -> if (!v1.contentEquals(v2)) return false
            v1 is CharArray && v2 is CharArray -> if (!v1.contentEquals(v2)) return false
            v1 is BooleanArray && v2 is BooleanArray -> if (!v1.contentEquals(v2)) return false
            v1 is UByteArray && v2 is UByteArray -> if (!v1.contentEquals(v2)) return false
            v1 is UShortArray && v2 is UShortArray -> if (!v1.contentEquals(v2)) return false
            v1 is UIntArray && v2 is UIntArray -> if (!v1.contentEquals(v2)) return false
            v1 is ULongArray && v2 is ULongArray -> if (!v1.contentEquals(v2)) return false
            else ->
        }

    }
    return true
}

fun <T> T?.contentDeepToString(processed: MutableList<T>): String {
    if (this in processed) {
        return "[...")
    }
    processed.add(this)
    return buildString {
        contentDeepToStringInternal(this, mutableListOf())
    }
}

fun <T> T?.contentDeepToStringInternal(result: StringBuilder, processed: MutableList<T>) {
    if (this in processed) {
        result.append("[...")
        return
    }
    processed.add(this)
    result.append('[

    for (i in indices) {
        if (i != 0) {
            result.append(', ')
        }
        val element = this[i]
        when (element) {
            null -> result.append("null")
            is Array<*> ->
                element.contentDeepToStringInternal(result, processed)
            is ByteArray ->
                result.append(element.contentToString())
            is ShortArray ->
                result.append(element.contentToString())
            is IntArray ->
                result.append(element.contentToString())
            is LongArray ->
                result.append(element.contentToString())
            is FloatArray ->
                result.append(element.contentToString())
            is DoubleArray ->
                result.append(element.contentToString())
            is CharArray ->
                result.append(element.contentToString())
            is BooleanArray ->
                result.append(element.contentToString())
            is UByteArray ->
                result.append(element.contentToString())
            is UShortArray ->
                result.append(element.contentToString())
            is UIntArray ->
                result.append(element.contentToString())
            is ULongArray ->
                result.append(element.contentToString())
            else ->
        }
    }
    return result.append(']')
}
public fun <K, V> Map<K, V>.withDefault(defaultValue: (key: K) -> V): Map<K, V> =
    when (this) {
        is MapWithDefault ->
            this.map.withDefault(defaultValue)
        else -> MapWithDefaultImpl(this, defaultValue)
    }

/**
 * Returns a wrapper of this mutable map, having the implicit default value provided with the specified function
 * [defaultValue].
 */
@kotlin.jvm.JvmName("withDefaultMutable")
public fun <K, V> MutableMap<K, V>.withDefault(defaultValue: (key: K) -> V): MutableMap<K, V> =
    when (this) {
        is MutableMapWithDefault ->
            this.map.withDefault(defaultValue)
        else -> MutableMapWithDefaultImpl(this, defaultValue)
    }

private interface MapWithDefault<K, out V> : Map<K, V> {
    public val map: Map<K, V>
    public fun getOrImplicitDefault(key: K): V
}

private interface MutableMapWithDefault<K, V> : MutableMap<K, V>, MapWithDefault<K, V> {
    public override val map: Map<K, V>
}

private class MapWithDefaultImpl<K, out V>(public override val map: Map<K, V>, private val default: (key: K) -> V) : MapWithDefault<K, V> {
    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)
    override fun get(key: K): V? = map.get(key)
    override val keys: Set<K> get() = map.keys
    override val values: Collection<V> get() = map.values
    override val entries: Set<Map.Entry<K, V>> get() = map.entries

    override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })
}

private class MutableMapWithDefaultImpl<K, V>(public override val map: MutableMap<K, V>, private val default: (key: K) -> V) : MutableMapWithDefault<K, V> {
    override fun equals(other: Any?): Boolean = map.equals(other)
    override fun hashCode(): Int = map.hashCode()
    override fun toString(): String = map.toString()
    override val size: Int get() = map.size
    override fun isEmpty(): Boolean = map.isEmpty()
    override fun containsKey(key: K): Boolean = map.containsKey(key)
    override fun containsValue(value: @UnsafeVariance V): Boolean = map.containsValue(value)
    override fun get(key: K): V? = map.get(key)
    override val keys: MutableSet<K> get() = map.keys
    override val values: MutableCollection<V> get() = map.values
    override val entries: MutableSet<MutableMap.MutableEntry<K, V>> get() = map.entries

    override fun put(key: K, value: V): V? = map.put(key, value)
    override fun remove(key: K): V? = map.remove(key)
    override fun putAll(from: Map<out K, V>): V? = map.putAll(from)
    override fun clear() = map.clear()

    override fun getOrImplicitDefault(key: K): V = map.getOrElseNullable(key, { default(key) })
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("CollectionsKt")

package kotlin.collections

import kotlin.random.Random

/**
 * Removes a single instance of the specified element from this collection, if it is present.
 */
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<T>.remove(element: T): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).remove(element)

/**
 * Removes all of this collection's elements that are also contained in the specified collection.
 */
@kotlin.internal.InlineOnly
public inline fun <@kotlin.internal.OnlyInputTypes T> MutableCollection<T>.removeAll(elements: Collection<T>): Boolean =
    @Suppress("UNCHECKED_CAST") (this as MutableCollection<T>).removeAll(elements)
@return `true` if any element was removed from the collection, `false` if the collection was not modified.

```kotlin
@Suppress("UNCHECKED_CAST")
@kotlin.internal.InlineOnly
public inline fun <T> MutableList<T>.retainAll(elements: Collection<T>): Boolean =
    elements.iterator().hasNext() && elements.iterator().forEachRemaining {
        if (remove(it)) retainAll(elements)
    }
```

### Public operators

#### `operator fun <T> MutableList<T>.plusAssign(element: T)`: Adds a single element to the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.plusAssign(element: T) {
    add(element)
}
```

#### `operator fun <T> MutableList<T>.plusAssign(elements: Collection<T>)`: Adds all elements to the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.plusAssign(elements: Collection<T>) {
    addAll(elements)
}
```

#### `operator fun <T> MutableList<T>.plusAssign(elements: Array<T>)`: Adds all elements to the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.plusAssign(elements: Array<T>) {
    addAll(elements)
}
```

#### `operator fun <T> MutableList<T>.plusAssign(elements: Sequence<T>)`: Adds all elements to the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.plusAssign(elements: Sequence<T>) {
    addAll(elements)
}
```

#### `operator fun <T> MutableList<T>.minusAssign(element: T)`: Removes an element from the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.minusAssign(element: T) {
    remove(element)
}
```

#### `operator fun <T> MutableList<T>.minusAssign(elements: Collection<T>)`: Removes all elements from the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.minusAssign(elements: Collection<T>) {
    removeAll(elements)
}
```

#### `operator fun <T> MutableList<T>.minusAssign(elements: Array<T>)`: Removes all elements from the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.minusAssign(elements: Array<T>) {
    removeAll(elements)
}
```

#### `operator fun <T> MutableList<T>.minusAssign(elements: Sequence<T>)`: Removes all elements from the mutable collection.

```kotlin
@kotlin.internal.InlineOnly
public inline operator fun <T> MutableList<T>.minusAssign(elements: Sequence<T>) {
    removeAll(elements)
}
```

### Public functions

#### `addAll(elements: Collection<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.addAll(elements: Collection<T>): Boolean {
    when (elements) {
        is Collection ->
            return addAll(elements)
        else -> {
            var result: Boolean = false
            for (item in elements)
                if (add(item)) result = true
            return result
        }
    }
}
```

#### `addAll(elements: Sequence<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.addAll(elements: Sequence<T>): Boolean {
    var result: Boolean = false
    for (item in elements) {
        if (add(item)) result = true
    }
    return result
}
```

#### `addAll(elements: Array<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.addAll(elements: Array<T>): Boolean {
    return addAll(elements.asList())
}
```

#### `removeAll(elements: Collection<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.removeAll(elements: Collection<T>): Boolean {
    return removeAll(elements.convertToSetForSetOperationWith(this))
}
```

#### `removeAll(elements: Array<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.removeAll(elements: Array<T>): Boolean {
    if (elements.isNotEmpty())
        return removeAll(elements.toHashSet())
    else
        return removeAllNothing()
}
```

#### `removeAll(elements: Sequence<T>): Boolean`:

```kotlin
public fun <T> MutableList<T>.removeAll(elements: Sequence<T>): Boolean {
    val set = elements.toHashSet()
    if (set.isNotEmpty())
        return removeAll(set)
    else
        return removeAllNothing()
}
```

#### `removeAll(elements: Collection<T>): Boolean`:

```kotlin
private fun MutableList<*>.retainNothing(): Boolean {
    val result = isNotEmpty()
    clear()
    return
}
```

#### `removeAll(elements: Array<T>): Boolean`:

```kotlin
private fun MutableList<*>.removeAllNothing(): Boolean {
    return removeAllNothing()
}
```

#### `removeAll(elements: Sequence<T>): Boolean`:

```kotlin
private fun MutableList<*>.removeAllNothing(): Boolean {
    return removeAllNothing()
}
```
result

* Removes all elements from this [MutableIterable] that match the given [predicate].

@return `true` if any element was removed from this collection, or `false` when no elements were removed and collection was not modified.

** Note: This method is marked as `@deprecated` in Kotlin 1.2.

private fun <T> MutableList<T>.filterInPlace(predicate: (T) -> Boolean, predicateResultToRemove: Boolean): Boolean {
    var writeIndex: Int = 0
    for (readIndex in 0..lastIndex) {
        val element = this[readIndex]
        if (predicate(element) == predicateResultToRemove) continue
        if (writeIndex != readIndex) this[writeIndex] = element
        writeIndex++
    }
    if (writeIndex < size) {
        for (removeIndex in lastIndex downTo writeIndex)
            removeAt(removeIndex)
        return true
    } else {
        return false
    }
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
package kotlin.collections

private open class ReversedListReadOnly<out T>(private val delegate: List<T>) : AbstractList<T>() {
    override val size: Int get() = delegate.size
    override fun get(index: Int): T = delegate[reverseElementIndex(index)]
}

private class ReversedList<T>(private val delegate: MutableList<T>) : AbstractMutableList<T>() {
    override val size: Int get() = delegate.size
    override fun get(index: Int): T = delegate[reverseElementIndex(index)]

    override fun clear() = delegate.clear()
    override fun removeAt(index: Int): T = delegate.removeAt(reverseElementIndex(index))

    override fun set(index: Int, element: T): T = delegate.set(reverseElementIndex(index), element)
    override fun add(index: Int, element: T) {
        delegate.add(reversePositionIndex(index), element)
    }
}

private fun List<*>::reverseElementIndex(index: Int) =
    if (index in 0..lastIndex) lastIndex - index else throw IndexOutOfBoundsException("Element index $index must be in range [0..$lastIndex].")

private fun List<*>::reversePositionIndex(index: Int) =
    if (index in 0..size) size - index else throw IndexOutOfBoundsException("Position index $index must be in range [0..$size].")

/**
 * Returns a reversed read-only view of the original List.
 * All changes made in the original list will be reflected in the reversed one.
 * @sample samples.collections.ReversedViews.asReversedList
 */
public fun <T> List<T>.asReversed(): List<T> = ReversedListReadOnly(this)

/**
 * Returns a reversed mutable view of the original mutable List.
 * All changes made in the original list will be reflected in the reversed one and vice versa.
 * @sample samples.collections.ReversedViews.asReversedMutableList
 */
@kotlin.jvm.JvmName("asReversedMutable")
public fun <T> MutableList<T>.asReversed(): MutableList<T> = ReversedList(this)

/**
 * Builds a [Sequence] lazily yielding values one by one.
 *
 * @see kotlin.sequences.generateSequence
 *
 * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
 * @sample samples.collections.Iterables.Building.iterable
 */
@SinceKotlin("1.3")
public fun <T> sequence(@BuilderInference block: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(block)
}

@SinceKotlin("1.3")
@Deprecated("Use 'sequence { }' function instead.", level = DeprecationLevel.ERROR)
@file:kotlin.internal.InlineOnly
public inline fun <T> buildSequence(@BuilderInference noinline builderAction: suspend SequenceScope<T>.() -> Unit): Sequence<T> = Sequence {
    iterator(builderAction)
}

/**
 * The scope for yielding values of a [Sequence] or an [Iterator], provides [yield] and [yieldAll] suspension functions.
 * @see sequence
 * @see iterator
 */
@RestrictsSuspension
@SinceKotlin("1.3")
public abstract class SequenceScope<in T> internal constructor() {
    /**
     * Yields a value to the [Iterator] being built and suspends it until the next value is requested.
     * @param value the value to yield
     * @param nextStep the next step of the inner iteration
     * @return the new sequence or iterator
     */
    public abstract suspend fun yield(value: T, nextStep: SequenceScope<in T>.NextStep) =

T)\n /**
 * Yields all values from the 'iterator' to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
 * The sequence of values returned by the given iterator can be potentially infinite.
 */
 public abstract suspend fun yieldAll(iterator: Iterator<T>)

    /**
     * Yields all values from the `iterator` to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
     * The sequence of potentially infinite sequence of values to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
     * The sequence can be potentially infinite.
     */
    public abstract suspend fun yieldAll(elements: Iterable<T>)

    /**
     * Yields potentially infinite sequence of values to the [Iterator] being built and suspends until all these values are iterated and the next one is requested.
     * @sample samples.collections.Sequences.Building.buildSequenceYieldAll
     */
    public suspend fun yieldAll(sequence: Sequence<T>) = yieldAll(sequence.iterator())

    @Deprecated("Use SequenceScope class instead.", ReplaceWith("SequenceScope<T>"))
    public typealias SequenceBuilder<T> = SequenceScope<T>

    private typealias State = Int

    private const val State_NotReady: State = 0
    private const val State_ManyNotReady: State = 1
    private const val State_ManyReady: State = 2
    private const val State_Ready: State = 3
    private const val State_Done: State = 4
    private const val State_Failed: State = 5

    private class SequenceBuilderIterator<T> : SequenceScope<T>(), Iterator<T>, Continuation<Unit> {
        private var state = State_NotReady
        private var nextValue: T? = null
        private var nextIterator: Iterator<T>? = null
        var nextStep: Continuation<Unit>? = null

        override fun hasNext(): Boolean {
            while (true) {
                when (state) {
                    State_NotReady -> {}
                    State_ManyNotReady ->
                        if (nextIterator!!.hasNext()) {
                            state = State_ManyReady
                            return true
                        } else {
                            nextIterator = null
                        }
                    State_Done -> return false
                    State_Ready, State_ManyReady -> return true
                    else -> throw exceptionalState()
                }

                state = State_Failed
                val step = nextStep!!
                nextStep = null
                resume(Unit)
            }
        }

        override fun next(): T {
            when (state) {
                State_NotReady, State_ManyNotReady -> return nextNotReady()
                State_ManyReady -> {
                    state = State_ManyNotReady
                    return nextIterator!!.next()
                }
                State_Ready -> {
                    state = State_NotReady
                    @Suppress("UNCHECKED_CAST")
                    val result = nextValue as T
                    nextValue = null
                    return result
                }
                else -> throw exceptionalState()
            }
        }

        private fun nextNotReady(): T {
            if (!hasNext()) throw NoSuchElementException() else return next()
        }

        private fun exceptionalState(): Throwable = when (state) {
            State_Done -> NoSuchElementException()
            State_Failed -> IllegalStateException("Iterator has failed.")
            else -> IllegalStateException("Unexpected state of the iterator: $state")
        }

        override suspend fun yield(value: T) {
            nextValue = value
            state = State_Ready
            return suspendCoroutineUninterceptedOrReturn { c ->
                nextStep = c
                COROUTINE_SUSPENDED
            }
        }

        override suspend fun yieldAll(iterator: Iterator<T>) {
            if (!iterator.hasNext()) return
            nextIterator = iterator
            state = State_ManyReady
            return suspendCoroutineUninterceptedOrReturn { c ->
                nextStep = c
                COROUTINE_SUSPENDED
            }
        }
    }

    // Completion continuation implementation
    override fun resumeWith(result: Result<Unit>) {
        result.getOrThrow() // just rethrow exception if it is there
        override suspend val value: T = null
        coroutineName = "windowedException($c)"
        value = nextValue
        state = State_Ready
        return suspendCoroutineUninterceptedOrReturn { c ->
            nextStep = c
            COROUTINE_SUSPENDED
        }
    }

    internal fun checkWindowSizeStep(size: Int, step: Int) {
        require(size > 0 && step > 0) {
            "Both size $size and step $step must be greater than zero."
        }
    }

        return sequence { windowedIterator(iterator(), size, step, partialWindows, reuseBuffer) }
    }

        return sequence { windowedIterator(iterator(), size, step, partialWindows, reuseBuffer) }
    }

    // Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
    // Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
inline fun Int.forward(n: Int): Int = (this + n) % capacity

@ExperimentalUnsignedTypes\private fun partition(array: UByteArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\private fun quickSort(array: UByteArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// UShortArray
@ExperimentalUnsignedTypes\private fun partition(array: UShortArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\private fun quickSort(array: UShortArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// UIntArray
@ExperimentalUnsignedTypes\private fun partition(array: UIntArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\private fun quickSort(array: UIntArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// ULongArray
@ExperimentalUnsignedTypes\private fun partition(array: ULongArray, left: Int, right: Int): Int {
    var i = left
    var j = right
    val pivot = array[(left + right) / 2]
    while (i <= j) {
        while (array[i] < pivot)
            i++
        while (array[j] > pivot)
            j--
        if (i <= j) {
            val tmp = array[i]
            array[i] = array[j]
            array[j] = tmp
            i++
            j--
        }
    }
    return i
}

@ExperimentalUnsignedTypes\private fun quickSort(array: ULongArray, left: Int, right: Int) {
    val index = partition(array, left, right)
    if (left < index - 1)
        quickSort(array, left, index - 1)
    if (index < right)
        quickSort(array, index, right)
}

// Interfaces
interface SortArray

/**
 * Sorts the given array using qsort algorithm.
 */
internal fun sortArray(array: UByteArray)    = quickSort(array, 0, array.size - 1)
internal fun sortArray(array: UShortArray)   = quickSort(array, 0, array.size - 1)
internal fun sortArray(array: UIntArray)     = quickSort(array, 0, array.size - 1)
internal fun sortArray(array: ULongArray)    = quickSort(array, 0, array.size - 1)

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
using the compiler argument `-Xopt-in=kotlin.contracts.ExperimentalContracts`.

```kotlin
@Suppress(
    "DEPRECATION",
    "ANY\n    @Retention(AnnotationRetention.BINARY)\n    @SinceKotlin("1.3")\n    @Experimental\n    @RequiresOptIn\n    @MustBeDocumented\n)\n
public annotation class ExperimentalContracts

/**
 * Provides a scope, where the functions of the contract DSL, such as [returns], [callsInPlace], etc., \n * can be used to describe the contract of a function.\n * This type is used as a receiver type of the lambda function passed to the [contract] function.\n * \n * # @see contract
 */
@ContractsDsl
@ExperimentalContracts
@SinceKotlin("1.3")
public interface ContractBuilder {
    /**
     * Describes a situation when a function returns normally, without any exceptions thrown.\n     * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.\n     * \n     * @see sample samples.contracts.returnsContract
     */
    @ContractsDsl public fun returns(): Returns

    /**
     * Describes a situation when a function returns normally with the specified return [value].\n     * The possible values of [value] are limited to `true`, `false` or `null`.\n     * Use [SimpleEffect.implies] function to describe a conditional effect that happens in such case.\n     */
    @ContractsDsl public fun returns(value: Any?): Returns

    /**
     * Specifies how many times a function invokes its function parameter in place.\n     */
    @ContractsDsl enum class InvocationKind {
        /**
         * A function parameter will be invoked one time or not invoked at all.\n         * A function parameter will be invoked one or more times.\n         * A function parameter is called in place, but it's unknown how many times it can be called.\n         * \n         * @param builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.\n         */
        AT_MOST_ONCE, AT_LEAST_ONCE, EXACTLY_ONCE, UNKNOWN
    }

    /**
     * Specifies the contract parameter [lambda] is invoked in place.\n     * \n     * @param builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.\n     */
    @ContractsDsl
    @InlineOnly
    @param builder the lambda where the contract of a function is described with the help of the [ContractBuilder] members.
```
SinceKotlin("1.3")
@Suppress("UNUSED_PARAMETER")
public inline fun contract(builder: ContractBuilder.() -> Unit) {
    /*
     * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
     * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
     */

    @SinceKotlin("1.3")
    public interface ContinuationInterceptor : CoroutineContext.Element {
        /**
         * The key that defines *the* context interceptor.
         */
        companion object Key :
            CoroutineContext.Key<ContinuationInterceptor>

        /**
         * Returns continuation that wraps the original [continuation], thus intercepting all resumptions.
         * This function is invoked by coroutines framework when needed and the resulting continuations are cached internally per each instance of the original [continuation].
         * This function may simply return original [continuation] if it does not want to intercept this particular continuation.
         * When the original [continuation] completes, coroutine framework invokes `releaseInterceptedContinuation` with the resulting continuation if it was intercepted, that is if `interceptContinuation` had previously returned a different continuation instance.
         * public fun <T> interceptContinuation(continuation: Continuation<T>): Continuation<T>
         * releaseInterceptedContinuation(continuation: Continuation<T>)
         */

            // getPolymorphicElement specialized for ContinuationInterceptor key
            @OptIn(ExperimentalStdlibApi::class)
            if (key is AbstractCoroutineContextKey<*, *>) {
                @Suppress("UNCHECKED_CAST")
                return if (key.isSubKey(this.key)) key.tryCast(this) as? E else null
            }
            @Suppress("UNCHECKED_CAST")
            return if (ContinuationInterceptor === key) this as E else null
        }

        public override fun minusKey(key: CoroutineContext.Key<*>): CoroutineContext =
            if (context === EmptyCoroutineContext) this else // fast path -- avoid lambda creation
            context.fold(this) { acc, element ->
                val removed =
acc.minusKey(element.key)
                if (removed === EmptyCoroutineContext) element else {
                    // make sure interceptor is always last in the context (and thus is fast to get when present)
                    val interceptor = removed[ContinuationInterceptor]
                    if (interceptor == null) CombinedContext(removed, element) else
                        val left = removed.minusKey(ContinuationInterceptor)
                    if (left === EmptyCoroutineContext) CombinedContext(element, interceptor) else

    /**
     * Returns a context containing elements from this context, but without an element with
     * the specified [key].
     */
    public fun minusKey(key: Key<*>): CoroutineContext

    /**
     * Key for the elements of
     * [CoroutineContext]. [E] is a type of element with this key.
     */
    public interface Key<E : Element>

    /**
     * An element of the [CoroutineContext]. An element of the coroutine context is a singleton context by itself.
     */
    public interface Element : CoroutineContext {
        /**
         * A key of this coroutine context element.
         */
        public val key: Key<*>

        public override operator fun <E : Element> get(key: Key<E>): E? =
            if (this.key == key) this as E else null

        public override fun <R> fold(initial: R, operation: (R, Element) -> R): R =
            operation(initial, this)

        public override fun minusKey(key: Key<*>): CoroutineContext =
            if (this.key == key) EmptyCoroutineContext else this
    }

    /**
     * Base class for [CoroutineContext.Element] implementations.
     */
    public abstract class AbstractCoroutineContextElement(public override val key:
        Key<*>) : Element

    /**
     * Base class for [CoroutineContext.Key] associated with polymorphic
     * [CoroutineContext.Element] implementation.
     * Polymorphic element implementation implies delegating its
     * [get] and [minusKey] to [getPolymorphicElement] and
     * [minusPolymorphicKey] respectively.
     */
    public abstract class AbstractCoroutineContextKey<B : Element, E : B>(
        baseKey: Key<B>,
        private val safeCast: (element: Element) -> E?
    ) : Key<E> {
        private val topmostKey: Key<*> = if (baseKey is
            AbstractCoroutineContextKey<*, *>) baseKey.topmostKey else baseKey

        internal fun tryCast(element: Element): E? = safeCast(element)
        internal fun isSubKey(key: Key<*>): Boolean = key === this || topmostKey
            === key

        /**
         * Returns the current element is it is a subtype of [E] or `null` otherwise.
         * @param key instance of base key
         * @param B base class of a polymorphic element
         * @param E element type associated with the current key
         * @param safeCast a function that can safely cast
         * abstract [CoroutineContext.Element] to the concrete [E] type
         */
        public fun E? = getPolymorphicElement(key)

        companion object Key : AbstractCoroutineContextKey<B, E>(Key, { it as
            DerivedElement })

        /**
         * Returns BaseElement?, non-null both for BaseElement and DerivedElement
         * instances
         */
        public fun E?: @param B base class of a polymorphic element
         * @param key instance of base key
         * @param E element type associated with the current key
         * @param safeCast a function that can safely cast
         * abstract [CoroutineContext.Element] to the concrete [E] type
         */
        public fun E? = getPolymorphicElement(key)

        companion object Key : AbstractCoroutineContextKey<E, E>(Key, { it as
            DerivedElement })

        /**
         * Returns BaseElement?, non-null only for DerivedElement
         * instance
         */
        public fun E?: @param E element type associated with the current key
         * @param safeCast a function that can safely cast
         * abstract [CoroutineContext.Element] to the concrete [E] type
         */
        public fun E? = getPolymorphicElement(key)
@Suppress("UNCHECKED_CAST")
        return if (key.isSubKey(this.key)) key.tryCast(this) as? E else null
    }
    @Suppress("UNCHECKED_CAST")
    return if (this.key === key) this as E else null
}

/**
 * Returns empty coroutine context if the element is associated with the given [key] in a polymorphic manner or `null` otherwise.
 * This method returns empty context if either [Element.key] is equal to the given [key] or if the [key] is associated with [Element.key] via [AbstractCoroutineContextKey].
 * See [AbstractCoroutineContextKey] for the example of usage.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Element.minusPolymorphicKey(key: Key<*>): CoroutineContext {
    if (key is AbstractCoroutineContextKey<*, *>) {
        return if (key.isSubKey(this.key) && key.tryCast(this) != null) EmptyCoroutineContext else this
    }
    return if (this.key === key) EmptyCoroutineContext else this
}

/**
 * An empty coroutine context.
 */
@SinceKotlin("1.3")
public object EmptyCoroutineContext : CoroutineContext, Serializable {
    private const val serialVersionUID: Long = 0
    private fun readResolve(): Any = EmptyCoroutineContext

    public override fun <E : Element> get(key: Key<E>): E? = null
    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R = initial
    public override fun plus(context: CoroutineContext): CoroutineContext = context
    public override fun minusKey(key: Key<*>): CoroutineContext = this
    public override fun hashCode(): Int = 0
    public override fun toString(): String = "EmptyCoroutineContext"
}

//--------------------- internal impl ---------------------

// this class is not exposed, but is hidden inside implementations/this is a left-biased list, so that `plus` works naturally
@SinceKotlin("1.3")
internal class CombinedContext(
    private val left: CoroutineContext,
    private val element: Element
) : CoroutineContext, Serializable {

    override fun <E : Element> get(key: Key<E>): E? {
        var cur = this
        while (true) {
            cur.element[key]?.let { return it }
            val next = cur.left
            if (next is CombinedContext) {
                cur = next
            } else {
                return next[key]
            }
        }
    }

    public override fun <R> fold(initial: R, operation: (R, Element) -> R): R =
        operation(left.fold(initial, operation), element)

    public override fun minusKey(key: Key<*>): CoroutineContext {
        element[key]?.let { return left }
        val newLeft = left.minusKey(key)
        return when {
            newLeft === left -> this
            newLeft === EmptyCoroutineContext -> element
            else ->
                CombinedContext(newLeft, element)
        }
    }

    private fun size(): Int {
        var cur = this
        var size = 2
        while (true) {
            cur = cur.left as? CombinedContext ?: return size
            size++
        }
    }

    private fun contains(element: Element): Boolean =
        get(element.key) == element

    private fun containsAll(context: CombinedContext): Boolean {
        var cur = context
        while (true) {
            if (!contains(cur.element)) return false
            val next = cur.left
            if (next is CombinedContext) {
                cur = next
            } else {
                return contains(next as Element)
            }
        }
    }

    override fun equals(other: Any?): Boolean =
        this === other || other is CombinedContext && other.size() == size() &&
        other.containsAll(this)

    override fun hashCode(): Int = left.hashCode() + element.hashCode()

    override fun toString(): String =
        "[" + fold("") { acc, element ->
            if (acc.isEmpty()) element.toString() else
            "$acc, $element"
        } + "]"

    private fun writeReplace(): Any {
        val n = size()
        val elements = arrayOfNulls<CoroutineContext>(n)
        var index = 0
        fold(Unit) { _, element -> elements[index++] = element }
        check(index == n)
        @Suppress("UNCHECKED_CAST")
        return Serialized(elements as Array<CoroutineContext>)
    }

    private class Serialized(val elements: Array<CoroutineContext>) :
        Serializable {
        companion object {
            private const val serialVersionUID: Long = 0L
        }

        private fun readResolve(): Any = elements.fold(EmptyCoroutineContext, CoroutineContext::plus)
    }
}
some moment in the\n * future when the result becomes available to resume the computation.\n * Otherwise, the return value of the [block] must have a type assignable to [T] and represents the result of this suspend function.\n * It means that the execution was not suspended and the [Continuation] provided to the [block] shall not be invoked.\n * As the result type of the [block] is declared as `Any?` and cannot be correctly type-checked,\n * its proper return type remains on the conscience of the suspend function's author.\n * Invocation of [Continuation.resumeWith] resumes coroutine directly in the invoker's thread without going through the\n * [Continuation.intercepted] function return value (including, but not limited to,\n * storing this value in other properties, returning it from other functions, etc).\n * It is implemented as property with getter to avoid ProGuard <clinit> problem with multifile IntrinsicsKt class.\n
```java
@SinceKotlin("1.3")
@Suppress("UNUSED_PARAMETER", "RedundantSuspendModifier")
public suspend inline fun <T> suspendCoroutineUninterceptedOrReturn(crossinline block: (Continuation<T>) -> Any?): T =
    throw NotImplementedError("Implementation of suspendCoroutineUninterceptedOrReturn is intrinsic\")\n
/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n */

package kotlin.experimental

/** Performs a bitwise AND operation between the two values.\n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.and(other: Byte): Byte =
    (this.toInt() and other.toInt()).toByte()\n
/** Performs a bitwise OR operation between the two values.\n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.or(other: Byte): Byte =
    (this.toInt() or other.toInt()).toByte()\n
/** Performs a bitwise XOR operation between the two values.\n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline infix fun Byte.xor(other: Byte): Byte =
    (this.toInt() xor other.toInt()).toByte()\n
/** Inverts the bits in this value.\n */
@SinceKotlin("1.1")
@kotlin.internal.InlineOnly
public inline fun Byte.inv(): Byte =
    (this.toInt().inv()).toByte()\n```

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 2480
Inference augmenting annotations.

* Any usage of a declaration annotated with `@ExperimentalTypeInference` must be accepted either by annotating that usage with the `[OptIn]` annotation, e.g. `@OptIn(ExperimentalTypeInference::class)`, or by using the compiler argument `-Xopt-in=kotlin.experimental.ExperimentalTypeInference`.

* Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.

* Specifies that the corresponding type should be ignored during type inference.

* Specifies that the constraint built for the type during type inference should be an equality one.

* Specifies that the corresponding member has the lowest priority in overload resolution.

* The value of this type parameter should be mentioned in input types (argument types, receiver type or expected type).

* Specifies that this function should not be called directly without inlining.

* The value of this parameter should be a property reference expression (`this::foo`), referencing a `lateinit` property.

* Note that prior to 1.2, only [RequireKotlinVersionKind.LANGUAGE_VERSION] was supported, so the Kotlin compiler before 1.2 is going to treat any [RequireKotlin] as if it requires the language version. Since 1.2, the Kotlin compiler supports [RequireKotlinVersionKind.LANGUAGE_VERSION], [RequireKotlinVersionKind.COMpiler_VERSION] and [RequireKotlinVersionKind.API_VERSION]. If the actual value of [versionKind] is something different (e.g. a new version kind, added in future versions of Kotlin), Kotlin 1.2 is going to ignore this [RequireKotlin] altogether, where as Kotlin before 1.2 is going to treat this as a
requirement
* on the language version.

* This annotation is erased at compile time; its arguments are stored in a more compact form in the Kotlin metadata.

* @Target(AnnotationTarget.CLASS, AnnotationTarget.FUNCTION, AnnotationTarget.PROPERTY, AnnotationTarget.CONSTRUCTOR, AnnotationTarget.TYPEALIAS)

* @Retention(AnnotationRetention.SOURCE)

* @Repeatable

* The kind of the version that is required by [RequireKotlin].

* * SinceKotlin("1.2")

* internal annotation class RequireKotlin(val version: String, val message: String = "", val level: DeprecationLevel = DeprecationLevel.ERROR)

* val versionKind: RequireKotlinVersionKind = RequireKotlinVersionKind.LANGUAGE_VERSION

* val errorCode: Int = -1

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

* * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
this callback is invoked.  * If the callback returns `true` the value of the property is being set to the new value,  * and if the callback returns `false` the new value is discarded and the property remains its old value.  *

```kotlin
protected open fun beforeChange(property: KProperty<*>, oldValue: T, newValue: T): Boolean = true

* The callback which is called after the change of the property is made. The value of the property  * has already been changed when this callback is invoked.

```
0xFFFF_FFFF\n
else -> {\n
val bitCount = fastLog2(nHigh)\n
nextBits(bitCount).toLong().shl(32) + nextInt()\n
}\n
}\n
} else {\n
var v:
Long\n
do {\n
val bits = nextLong().ushr(1)\n
v = bits % n\n
} while (bits
- v + (n - 1) < 0)\n
rnd = v\n
}\n
return from + rnd\n
} else {\n
while (true) {\n
val rnd = nextLong()\n
if (rnd in from until until) return rnd\n
}\n
}\n }\n\n /**\n *
Gets the next random [Boolean] value.\n */\n public open fun nextBoolean(): Boolean = nextBits(1) != 0\n\n
/**\n * Gets the next random [Double] value uniformly distributed between 0 (inclusive) and 1 (exclusive).\n
*/\n public open fun nextDouble(): Double = doubleFromParts(nextBits(26), nextBits(27))\n\n /**\n * Gets
the next random non-negative `Double` from the random number generator less than the specified [until] bound.\n
*\n * Generates a `Double` random value uniformly distributed between 0 (inclusive) and [until] (exclusive).\n
*\n * @throws IllegalArgumentException if [until] is negative or zero.\n */\n public open fun
nextDouble(until: Double): Double = nextDouble(0.0, until)\n\n /**\n * Gets the next random `Double` from
the random number generator in the specified range.\n *\n * Generates a `Double` random value uniformly
distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.\n *\n * [from] and [until]
must be finite otherwise the behavior is unspecified.\n *\n * @throws IllegalArgumentException if [from] is
greater than or equal to [until].\n */\n public open fun nextDouble(from: Double, until: Double): Double {\n
checkRangeBounds(from, until)\n
val size = until - from\n
val r = if (size.isInfinite() && from.isFinite()
&& until.isFinite()) {\n
val r1 = nextDouble() * (until / 2 - from / 2)\n
from + r1 + r1\n
} else {\n
from + nextDouble() * size\n
}\n
return if (r >= until) until.nextDown() else r\n }\n\n /**\n *
Gets the next random [Float] value uniformly distributed between 0 (inclusive) and 1 (exclusive).\n */\n public
open fun nextFloat(): Float = nextBits(24) / (1 shl 24).toFloat()\n\n /**\n * Fills a subrange of the specified byte
[array] starting from [fromIndex] inclusive and ending [toIndex] exclusive\n * with random bytes.\n *\n *
@return [array] with the subrange filled with random bytes.\n */\n public open fun nextBytes(array: ByteArray,
fromIndex: Int = 0, toIndex: Int = array.size): ByteArray {\n
require(fromIndex in 0..array.size && toIndex in
0..array.size) { \"fromIndex ($fromIndex) or toIndex ($toIndex) are out of range: 0..${array.size}.\" }\n
require(fromIndex <= toIndex) { \"fromIndex ($fromIndex) must be not greater than toIndex ($toIndex).\" }\n\n
val steps = (toIndex - fromIndex) / 4\n\n
var position = fromIndex\n
repeat(steps) {\n
val v =
nextInt()\n
array[position] = v.toByte()\n
array[position + 1] = v.ushr(8).toByte()\n
array[position + 2] = v.ushr(16).toByte()\n
array[position + 3] = v.ushr(24).toByte()\n
position += 4\n
}\n\n
val remainder = toIndex - position\n
val vr = nextBits(remainder * 8)\n
for (i in 0 until
remainder) {\n
array[position + i] = vr.ushr(i * 8).toByte()\n
}\n\n
return array\n }\n\n /**\n *
Fills the specified byte [array] with random bytes and returns it.\n *\n * @return [array] filled with random
bytes.\n */\n public open fun nextBytes(array: ByteArray): ByteArray = nextBytes(array, 0, array.size)\n\n
/**\n * Creates a byte array of the specified [size], filled with random bytes.\n */\n public open fun
nextBytes(size: Int): ByteArray = nextBytes(ByteArray(size))\n\n\n /**\n * The default random number
generator.\n *\n * On JVM this generator is thread-safe, its methods can be invoked from multiple threads.\n
*\n * @sample samples.random.Randoms.defaultRandom\n */\n companion object Default : Random() {\n\n
private val defaultRandom: Random = defaultPlatformRandom()\n\n
override fun nextBits(bitCount: Int): Int
= defaultRandom.nextBits(bitCount)\n
override fun nextInt(): Int = defaultRandom.nextInt()\n
override fun
nextInt(until: Int): Int = defaultRandom.nextInt(until)\n
override fun nextInt(from: Int, until: Int): Int =
defaultRandom.nextInt(from, until)\n\n
override fun nextLong(): Long = defaultRandom.nextLong()\n
override fun nextLong(until: Long): Long = defaultRandom.nextLong(until)\n
override fun nextLong(from:
Long, until: Long): Long = defaultRandom.nextLong(from, until)\n\n
override fun nextBoolean(): Boolean =
defaultRandom.nextBoolean()\n\n
override fun nextDouble(): Double = defaultRandom.nextDouble()\n
override fun nextDouble(until: Double): Double = defaultRandom.nextDouble(until)\n
override fun
nextDouble(from: Double, until: Double): Double = defaultRandom.nextDouble(from, until)\n\n
override fun
nextFloat(): Float = defaultRandom.nextFloat()\n\n
override fun nextBytes(array: ByteArray): ByteArray =
defaultRandom.nextBytes(array)\n
override fun nextBytes(size: Int): ByteArray =

Open Source Used In IOx-FogDirector-AC18 1.17.0 2484


override fun nextBytes(array: ByteArray): Int
{
    @Deprecated("Use Default companion object instead", level = DeprecationLevel.HIDDEN)
    @Suppress("DEPRECATION_ERROR")
    @kotlin.jvm.JvmField
}

    public object Companion : Random() {
        override fun nextBytes(bitCount: Int): ByteArray = defaultRandom.nextBytes(array, fromIndex, toIndex)

        @Deprecated("Use Default companion object instead", level = DeprecationLevel.HIDDEN)
        @Suppress("DEPRECATION_ERROR")
        @kotlin.jvm.JvmField
    }

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Int` value.
 */
    public fun Random(seed: Int): Random = XorWowRandom(seed, seed.shr(31))

/**
 * Returns a repeatable random number generator seeded with the given [seed] `Long` value.
 */
    public fun Random(seed: Long): Random = XorWowRandom(seed.toInt(), seed.shr(32).toInt())

internal expect fun defaultPlatformRandom(): Random
internal expect fun doubleFromParts(hi26: Int, low27: Int): Double

@UseExperimental(ExperimentalStdlibApi::class)
internal fun fastLog2(value: Int): Int = 31 - value.countLeadingZeroBits()

/** Takes upper [bitCount] bits (0..32) from this number. */
internal fun Int.takeUpperBits(bitCount: Int): Int =
    this.ushr(32 - bitCount) and (-bitCount).shr(31)

internal fun checkRangeBounds(from: Int, until: Int) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Long, until: Long) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun checkRangeBounds(from: Double, until: Double) = require(until > from) {
    boundsErrorMessage(from, until)
}

internal fun boundsErrorMessage(from: Any, until: Any) = "Range is empty: [$from, $until]."

@SinceKotlin("1.3")
public fun Random.nextInt(range: IntRange): Int = when {
    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")
    range.last < Int.MAX_VALUE -> nextInt(range.first, range.last + 1)
    range.first > Int.MIN_VALUE -> nextInt(range.first - 1, range.last) + 1
    else -> nextInt()
}

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(): UInt = nextInt().toUInt()

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(until: UInt): UInt = nextUInt(0u, until)

@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun Random.nextUInt(upperBound: UInt): UInt = nextUInt(until = upperBound)
range.
*undef* Generates a [UInt] random value uniformly distributed between the specified [from] (inclusive) and [until] (exclusive) bounds.
*undef* @throws IllegalArgumentException if [from] is greater than or equal to [until].
*undef@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextUInt(from: UInt, until: UInt):
    UInt \{ checkULongRangeBounds(from, until) \} val signedFrom = from.toInt() xor Int.MIN_VALUE \n    val signedUntil = until.toInt() xor Int.MIN_VALUE \n    return signedResult.toUInt() \n\}/*\*\* */ * Gets the next random [UInt] from the random number generator in the specified [range].\n* \* * Generates a [UInt] random value uniformly distributed in the specified [range]:\n* from \textquoteleft range.start\textquoteright inclusive to \textquoteleft range.endInclusive\textquoteright inclusive.\n* * \* @throws IllegalArgumentException if [range] is empty.\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextUInt(range: UIntRange): UInt = when {\n    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range: $range")\n    range.last < UInt.MAX_VALUE -> nextUInt(range.first, range.last + 1u)\n    range.first > UInt.MIN_VALUE -> nextUInt(range.first - 1u, range.last) + 1u\n    else -> nextUInt(\n\)}/*\*\* */ * Gets the next random [ULong] from the random number generator.\n* * * Generates a [ULong] random value uniformly distributed between [ULong.MIN_VALUE] and [ULong.MAX_VALUE] (inclusive).\n* * * @throws IllegalArgumentException if [from] is greater than or equal to [until].\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextULong(from: ULong, until: ULong): ULong {
    checkULongRangeBounds(from, until) \n    val signedFrom = from.toLong() xor Long.MIN_VALUE \n    val signedUntil = until.toLong() xor Long.MIN_VALUE \n    return signedResult.toULong() \n\}/*\*\* */ * Gets the next random [ULong] from the random number generator in the specified [range].\n* * * Generates a [ULong] random value uniformly distributed in the specified [range]:\n* from \textquoteleft range.start\textquoteright inclusive to \textquoteleft range.endInclusive\textquoteright inclusive.\n* * * @throws IllegalArgumentException if [range] is empty.\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextULong(range: ULongRange): ULong = when {\n    range.isEmpty() -> throw IllegalArgumentException("Cannot get random in empty range:
    $range")\n    range.last < ULong.MAX_VALUE -> nextULong(range.first, range.last + 1u)\n    range.first > ULong.MIN_VALUE -> nextULong(range.first - 1u, range.last) + 1u\n    else -> nextULong(\n\)}/*\*\* */ Fills the specified unsigned byte [array] with random bytes and returns it.\n* * * @return [array] filled with random bytes.\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextUBytes(array: UByteArray): UByteArray \n\{ nextBytes(array.asByteArray()) \} return array(\n\}/*\*\* */ Creates an unsigned byte array of the specified [size], filled with random bytes.\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextUBytes(size: Int): UByteArray = nextBytes(size).asUByteArray() \n\}/*\*\* */ Fills a subrange of the specified \textquoteleft UByte\textquoteright [array] starting from [fromIndex] inclusive and ending [toIndex] exclusive with random UBytes.\n* * * @return [array] with the subrange filled with random bytes.\n* */@SinceKotlin("1.3")@ExperimentalUnsignedTypes@n{\textbf{public} fun Random.nextUBytes(array: UByteArray, fromIndex: Int = 0, toIndex: Int = array.size): UByteArray \n\{ if nextBytes(array.asByteArray(), fromIndex, toIndex) \} return array(\n\}/*\*\* */ Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n* Use of this
source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

package kotlin.random

/**
 * Random number generator, using Marsaglia's xorwow algorithm
 * Cycles after 2^192 - 2^32 repetitions.
 */

internal class XorWowRandom
internal constructor(
    private var x: Int,
    private var y: Int,
    private var z: Int,
    private var w: Int,
    private var v: Int,
    private var addend: Int
) : Random() {

    internal constructor(seed1: Int, seed2: Int) :
            this(seed1, seed2, 0, 0,
                (seed1 shl 10) xor (seed2 ushr 4),
            addend += 362437
        v = t
    }

    override fun nextInt(): Int {
        // Equivalent to the xorxow algorithm
        var t = x
        t = t xor (t ushr 2)
        x = y
        y = z
        z = w
        val v0 = v
        w = v0
        t = (t xor (t shl 1)) xor v0 xor (v0 shl 4)
        v = t
        addend += 362437
        return t + addend
    }

    override fun nextBits(bitCount: Int): Int =
        nextInt().takeUpperBits(bitCount)
}

@file:kotlin.jvm.JvmMultifileClass
@file:kotlin.jvm.JvmName("RangesKt")

package kotlin.ranges

/**
 * Represents a range of Comparable values.
 */
private open class ComparableRange<T : Comparable<T>>(
    override val start: T,
    override val endInclusive: T
) : ClosedRange<T> {

    override fun equals(other: Any?): Boolean {
        return other is ComparableRange<*>
            && (isEmpty() && other.isEmpty() ||
                start == other.start && endInclusive == other.endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else 31 * start.hashCode() + endInclusive.hashCode()
    }

    override fun toString(): String = "$start..$endInclusive"
}

/**
 * Creates a range from this Comparable value to the specified that value.
 * This value needs to be smaller than that value, otherwise the returned range will be empty.
 * @sample samples.ranges.Ranges.rangeFromComparable
 */
public operator fun <T : Comparable<T>> T.rangeTo(that: T): ClosedRange<T> = ComparableRange(this, that)

/**
 * Represents a range of floating point numbers.
 * Extends ClosedRange interface providing custom operation lessThanOrEquals for comparing values of range domain type.
 * This interface is implemented by floating point ranges returned by Float.rangeTo and Double.rangeTo operators to achieve IEEE-754 comparison order instead of total order of floating point numbers.
 * Extends [ClosedRange] interface providing custom operation lessThanOrEquals for comparing values of range domain type.
 * This interface is implemented by floating point ranges returned by Float.rangeTo and Double.rangeTo operators to achieve IEEE-754 comparison order instead of total order of floating point numbers.
 */
@SinceKotlin("1.1")
public interface ClosedFloatingPointRange<T : Comparable<T>> : ClosedRange<T> {
    override fun contains(value: T): Boolean = lessThanOrEquals(start, value) && lessThanOrEquals(value, endInclusive)
    override fun isEmpty(): Boolean = !lessThanOrEquals(start, endInclusive)

    /**
     * Compares two values of range domain type and returns true if first is less than or equal to second.
     */
    fun lessThanOrEquals(a: T, b: T): Boolean
}

/**
 * A closed range of values of type `Double`.
 * Numbers are compared with the ends of this range according to IEEE-754.
 * @sample samples.ranges.ranges.RangeFromDouble
 */
private class ClosedDoubleRange(
    start: Double,
    endInclusive: Double
) : ClosedFloatingPointRange<Double> {
    private val _start = start
    private val _endInclusive = endInclusive

    override val start: Double get() = _start
    override val endInclusive: Double get() = _endInclusive

    override fun lessThanOrEquals(a: Double, b: Double): Boolean = a <= b

    override fun contains(value: Double): Boolean = value >= _start && value <= _endInclusive
    override fun isEmpty(): Boolean = !_start <= _endInclusive

    override fun equals(other: Any?): Boolean {
        return other is ClosedDoubleRange
            && (isEmpty() && other.isEmpty() ||
                _start == other._start && _endInclusive == other._endInclusive)
    }

    override fun hashCode(): Int {
        return if (isEmpty()) -1 else 31 * _start.hashCode() + _endInclusive.hashCode()
    }

    override fun toString(): String = "$_start..$_endInclusive"
}

/**
 * Creates a range from this Double value to the specified that value.
 * Numbers are compared with the ends of this range according to IEEE-754.
 * @sample samples.ranges.ranges.RangeFromDouble
 */
@SinceKotlin("1.1")
public operator fun Double.rangeTo(that: Double): ClosedFloatingPointRange<Double> = ClosedDoubleRange(this, that)
/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect

/**
 * Casts the given 
 [value][value] to the 
 [class][class] represented by this 
 [KClass][KClass] object.
 * Throws an exception if the value is `null` or if it is not an instance of
 * this class.
 * This is an experimental function that behaves as a similar function from kotlin.reflect.full on
 * JVM.
 * @see [KClass.isInstance]
 * @see [KClass.safeCast]
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@LowPriorityInOverloadResolution
fun <T : Any>
KClass<T>.cast(value: Any?): T {
    if (!isInstance(value)) throw ClassCastException("Value cannot be cast to
$qualifiedName")
    return value as T
}

/**
 * Casts the given 
 [value][value] to the class represented by this 
 [KClass][KClass] object.
 * Returns `null` if the value is `null` or if it is not an instance of
 * this class.
 * This is an experimental function that behaves as a similar function from kotlin.reflect.full on
 * JVM.
 * @see [KClass.isInstance]
 * @see [KClass.safeCast]
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@LowPriorityInOverloadResolution
fun <T : Any>
KClass<T>.safeCast(value: Any?): T? {
    return if (isInstance(value)) value as T else null
}

/**
 * Returns a runtime representation of the given reified type 
 [T][T] as an instance of 
 [KType][KType].
 * @see [KType]
 */
@Suppress("unused") // KT-12448
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun 
<reified T> typeOf(): KType =
    throw UnsupportedOperationException("This function is implemented as an
intrinsic on all supported platforms.")

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.reflect

/**
 * Returns a runtime representation of the given reified type 
 [T][T] as an instance of 
 [KType][KType].
 * @see [KType]
 */
@Suppress("unused") // KT-12448
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public inline fun 
<reified T> typeOf(): KType =
    throw UnsupportedOperationException("This function is implemented as an
intrinsic on all supported platforms.")

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

/**
 * An object to which char sequences and values can be appended.
 * @nexpect interface
 * Appendable
 * @param value the character sequence to append. 
 * Appendable
 * @param value the character sequence to append.
 * Appendable
 * Appendable
 * Appendable
 * Appendable
 * Appendable
 * Appendable
character sequence from which a subsequence is appended. If [value] is `null`, then characters are appended as if [value] contained the four characters `\"null\"`. * @param startIndex the beginning (inclusive) of the subsequence to append. * @param endIndex the end (exclusive) of the subsequence to append. * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`. * @param value the character sequence from which a subsequence is appended. If [value] is `null`, then characters are appended as if [value] contained the four characters `\"null\"`. * @param startIndex the beginning (inclusive) of the subsequence to append. * @param endIndex the end (exclusive) of the subsequence to append. * @throws IndexOutOfBoundsException or [IllegalArgumentException] when [startIndex] or [endIndex] is out of range of the [value] character sequence indices or when `startIndex > endIndex`. *
```kotlin
val lines = lines().filter { !it.isBlank() } .map { it.indentWidth().min() ?: 0f } return lines.reduce { newIndent, line ->
    line.drop(minCommonIndent) } , getIndentFunction(newIndent) , { line ->
        line.drop(minCommonIndent) }
    }

/**
 * Detects a common minimal indent of all the input lines, removes it from every line and
 * also removes the first and the last lines if they are blank (notice difference blank vs empty).
 * Note that blank lines do not affect the detected indent level. In case if there are non-blank lines with no leading
 * whitespace characters (no indent at all) then the common indent is 0, and therefore this function doesn't change
 * the indentation. Doesn't preserve the original line endings.
 */
public fun String.trimIndent(): String = replaceIndent"

/**
 * Detects a common minimal indent like it does [trimIndent] and replaces it with the specified [newIndent].
 */
public fun String.replaceIndent(newIndent: String = ""):
    lines().reindent { line ->
        line.drop(minCommonIndent) }

/**
 * Prepends [indent] to every line of the original string.
 * Doesn't preserve the original line endings.
 */
public fun String.prependIndent(indent: String = "    "):
    lineSequence().map { it.isNotBlank() -> it } .joinToString("\n")

private fun String.indentWidth(): Int = indexOfFirst { !it.isWhitespace() }.let { if (it == -1) length else it }

private fun getIndentFunction(indent: String) = when {
    indent.isEmpty() -> { line: String -> line }
    else -> { line: String -> indent + line }
}

private inline fun List<String>.reindent(resultSizeEstimate: Int, indentAddFunction: (String) -> String, indentCutFunction: (String) -> String?):
    mapIndexedNotNull { index, value ->
        if ((index == 0 || index == lastIndex) && value.isBlank())
            null
        else
            indentCutFunction(value) ?: indentAddFunction(value)
    }

.object Typography {
    /** The character " 
        \u2013 quotation mark */
    public const val quote: Char = '\u0022'
    /** The character $ 
        \u2013 dollar sign */
    public const val dollar: Char = '\u0024'
    /** The character & 
        \u2013 ampersand */
    public const val amp: Char = '\u0026'
    /** The character < 
        \u003C 
        \u2013 less-than sign */
    public const val less: Char = '\u003C'
    /** The character > 
        \u003E 
        \u2013 greater-than sign */
    public const val greater: Char = '\u003E'
    /** The non-breaking space character */
    public const val nbsp: Char = '\u00A0'
    /** The character × */
    public const val times: Char = '\u00D7'
    /** The character ¢ */
    public const val cent: Char = '\u00A2'
    /** The character £ */
    public const val pound: Char = '\u00A3'
    /** The character § */
    public const val section: Char = '\u00A7'
    /** The character © */
    public const val copyright: Char = '\u00A9'
    /** The character « */
    public const val leftGuillemete: Char = '\u00AB'
    /** The character » */
    public const val rightGuillemete: Char = '\u00BB'
    /** The character · */
    public const val middleDot: Char = '\u00B7'
    /** The character ½ */
    public const val half: Char = '\u00BD'
    /** The character \u00B1 */
    public const val plusMinus: Char = '\u00B1'
    /** The character \u00B7 */
    public const val middleDot: Char = '\u00B7'
    /** The character \u00B1 */
    public const val plusMinus: Char = '\u00B1'
    ```
public const val rightDoubleQuote: Char = '\u201D'
/** The character " */
public const val lowDoubleQuote: Char = '\u201E'
/** The character † */
public const val dagger: Char = '\u2020'
/** The character ‡ */
public const val doubleDagger: Char = '\u2021'
/** The character • */
public const val bullet: Char = '\u2022'
/** The character … */
public const val ellipsis: Char = '\u2026'
/** The character ′ */
public const val prime: Char = '\u2032'
/** The character ″ */
public const val doublePrime: Char = '\u2033'
/** The character € */
public const val euro: Char = '\u20AC'
/** The character ™ */
public const val tm: Char = '\u2122'
/** The character ≈ */
public const val almostEqual: Char = '\u2248'
/** The character ≠ */
public const val notEqual: Char = '\u2260'
/** The character ≤ */
public const val lessOrEqual: Char = '\u2264'
/** The character ≥ */
public const val greaterOrEqual: Char = '\u2265'
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin.text

/**
 * Represents a collection of captured groups in a single match of a regular expression.
 * This collection has size of `groupCount + 1` where `groupCount` is the count of groups in the regular expression.
 * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
 * An element of the collection at the particular index can be `null`, if the corresponding group in the regular expression is optional and there was no match captured by that group.
 */
public interface MatchGroupCollection : Collection<MatchGroup?> {

    /** Returns a group with the specified [index].
     * @return An instance of [MatchGroup] if the group with the specified [index] was matched or `null` otherwise.
     * Groups are indexed from 1 to the count of groups in the regular expression. A group with the index 0 corresponds to the entire match.
     */
    public operator fun get(index: Int): MatchGroup?
}

/**
 * Extends [MatchGroupCollection] by introducing a way to get matched groups by name, when regex supports it.
 */
@SinceKotlin("1.1")
public interface MatchNamedGroupCollection : MatchGroupCollection {
    /**
     * Returns a named group with the specified [name].
     * @return An instance of [MatchGroup] if the group with the specified [name] was matched or `null` otherwise.
     * @throws IllegalArgumentException if there is no group with the specified [name] defined in the regex pattern.
     * @throws UnsupportedOperationException if getting named groups isn't supported on the current platform.
     */
    public operator fun get(name: String): MatchGroup?
}

/**
 * Represents the results from a single regular expression match.
 */
public interface MatchResult {
    /** The range of indices in the original string where match was captured. */
    public val range: IntRange
    /** The substring from the input string captured by this match. */
    public val value: String
    /**
     * A collection of groups matched by the regular expression.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     */
    public val groups: MatchGroupCollection
    /**
     * A list of matched indexed group values.
     * Groups are indexed from 1 to `groupCount` and group with the index 0 corresponds to the entire match.
     * If the group in the regular expression is optional and there were no match captured by that group, corresponding item in [groupValues] is an empty string.
     */
    public val groupValues: List<String>

    /** Returns a new [MatchResult] with the results for the next match, starting at the position at which the last match ended (at the character after the last matched character).
     */
    public fun next(): MatchResult?

    /** Provides components for destructuring assignment of group values.
     * If the group in the regular expression is optional and there were no match captured by that group, corresponding component value is an empty string.
     */
    public val destructured: Destructured get() = Destructured(this)
}

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package IOx-FogDirector-AC18 1.17.0

/**
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

public class Destructured internal constructor(public val match: MatchResult) {
    @kotlin.internal.InlineOnly
    public operator inline fun component1(): String = match.groupValues[1]
    @kotlin.internal.InlineOnly
    public operator inline fun component2(): String = match.groupValues[2]
    @kotlin.internal.InlineOnly
    public operator inline fun component3(): String = match.groupValues[3]
    @kotlin.internal.InlineOnly
    public operator inline fun component4(): String = match.groupValues[4]
    @kotlin.internal.InlineOnly
    public operator inline fun component5(): String = match.groupValues[5]
    @kotlin.internal.InlineOnly
    public operator inline fun component6(): String = match.groupValues[6]
    @kotlin.internal.InlineOnly
    public operator inline fun component7(): String = match.groupValues[7]
    @kotlin.internal.InlineOnly
    public operator inline fun component8(): String = match.groupValues[8]
    @kotlin.internal.InlineOnly
    public operator inline fun component9(): String = match.groupValues[9]
    @kotlin.internal.InlineOnly
    public operator inline fun component10(): String = match.groupValues[10]

    /**
     * Returns destructured group values as a list of strings.
     * First value in the returned list corresponds to the value of the first group, and so on.
     * @sample samples.text.Regexps.matchDestructuringToGroupValues
     */
    public fun toList(): List<String> = match.groupValues.subList(1, match.groupValues.size)
}

@SinceKotlin("1.3")
@ExperimentalTime
@Suppress("NON_PUBLIC_PRIMARY_CONSTRUCTOR_O F_INLINE_CLASS")
public inline class Duration internal constructor(internal val value: Double) :
    Comparable<Duration> {
    // TODO: backend fails on init block, wait for KT-28055

    companion object {
        /** The duration equal to exactly 0 seconds. */
        public val ZERO: Duration = Duration(0.0)

        /** The duration whose value is positive infinity. It is
         * used to represent infinite timeouts. */
        public val INFINITE: Duration =
            Duration(Double.POSITIVE_INFINITY)

        /** Converts the given time duration 
         * expressed in the specified 
         * into the specified 
         * duration units. */
        public fun convert(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit): Double =
            convertDurationUnit(value, sourceUnit, targetUnit)

        // arithmetic operators

        /** Returns the negative of this value. */
        public operator fun unaryMinus(): Duration = Duration(-value)

        /** Returns a duration whose value is the sum of this and 
         * duration values. */
        public operator fun plus(other: Duration): Duration = Duration(value + other.value)

        /** Returns a duration whose value is this 
         * duration value multiplied by the given 
         * number. */
        public operator fun times(scale: Int): Duration =
            Duration(value * scale)

        /** Returns a duration whose value is this duration value multiplied by the given 
         * scale number. */
        public operator fun times(scale: Double): Duration =
            Duration(value * scale)

        /** Returns a number that is the ratio of this 
         * and [other] duration values. */
        public operator fun div(other: Duration): Double =
            Duration(value / other.value)

        /** Returns a duration whose value is the difference between this and [other] duration values. */
        public operator fun minus(other: Duration): Duration =
            Duration(value - other.value)

        /** Returns a duration whose value is this duration value multiplied by the given 
         * scale number. */
        public operator fun times(scale: Int): Duration =
            Duration(value * scale)

        /** Returns a duration whose value is this duration value multiplied by the given 
         * scale number. */
        public operator fun times(scale: Double): Duration =
            Duration(value * scale)

        /** Returns a number that is the ratio of this and [other] duration values. */
        public operator fun div(other: Duration): Double =
            Duration(value / other.value)

        // comparison operators

        /** Returns true, if the duration value is less than zero. */
        public fun isNegative(): Boolean = value < 0

        /** Returns true, if the duration value is greater than zero. */
        public fun isPositive(): Boolean = value > 0
    }

    // special accessors

    /** Returns the duration in hours. */
    public fun inHours(): Int =
        (value / DurationUnit.NANOSECONDS.inHours)

    /** Returns the duration in minutes. */
    public fun inMinutes(): Int =
        (value / DurationUnit.NANOSECONDS.inMinutes)

    /** Returns the duration in seconds. */
    public fun inSeconds(): Int =
        (value / DurationUnit.NANOSECONDS.inSeconds)

    /** Returns the duration in nanoseconds. */
    public fun inNanoseconds(): Int =
        (value / DurationUnit.NANOSECONDS.inNanoseconds)

    /** Returns the duration expressed in nans. */
    public fun inNanoseconds(): Int =
        (value / DurationUnit.NANOSECONDS.inNanoseconds)

    /** Converts a given time duration 
     * expressed in the specified 
     * into the specified 
     * duration units. */
    public fun convert(value: Double, sourceUnit: DurationUnit, targetUnit: DurationUnit):
        Duration =
            Duration(convertDurationUnit(value, sourceUnit, targetUnit))

    // properties

    /** Returns the duration represented by this value. */
    public val value: Double get() =
        (value / DurationUnit.NANOSECONDS.inNanoseconds)

    // lazy properties

    /** Returns the duration represented by this value. */
    public val hours: Int get() =
        (value / DurationUnit.NANOSECONDS.inHours)

    /** Returns the duration represented by this value. */
    public val minutes: Int get() =
        (value / DurationUnit.NANOSECONDS.inMinutes)

    /** Returns the duration represented by this value. */
    public val seconds: Int get() =
        (value / DurationUnit.NANOSECONDS.inSeconds)

    /** Returns the duration represented by this value. */
    public val nanoseconds: Int get() =
        (value / DurationUnit.NANOSECONDS.inNanoseconds)

    // extension funs

    /** Returns the duration represented by this value. */
    public fun toDuration():
        Duration =
            Duration(value / DurationUnit.NANOSECONDS.inNanoseconds)
public fun isPositive(): Boolean = value > 0

/** Returns true, if the duration value is infinite. */
public fun isInfinite(): Boolean = value.isInfinite()

/** Returns true, if the duration value is finite. */
public fun isFinite(): Boolean = value.isFinite()

/** Returns the absolute value of this value. The returned value is always non-negative. */
val absoluteValue: Duration get() = if (isNegative()) -this else this

override fun compareTo(other: Duration): Int = this.value.compareTo(other.value)

// splitting to components

/**
 * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration, and its absolute value is less than 24;
 * `days` represents the whole number of days in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
    action(inDays.toInt(), hoursComponent, minutesComponent, secondsComponent, nanosecondsComponent)

/**
 * Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
    action(inMinutes.toInt(), secondsComponent, nanosecondsComponent)

/**
 * Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
    action(inHours.toInt(), minutesComponent, secondsComponent, nanosecondsComponent)

/**
 * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int, seconds: Int) -> T): T =
    action(inDays.toInt(), hoursComponent, minutesComponent, secondsComponent)

/**
 * Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int) -> T): T =
    action(inDays.toInt(), hoursComponent, minutesComponent)

/**
 * Splits this duration into days, hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (seconds: Int, nanoseconds: Int) -> T): T =
    action(secondsComponent, nanosecondsComponent)

/**
 * Splits this duration into seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (minutes: Int, seconds: Int) -> T): T =
    action(minutesComponent, secondsComponent)

/**
 * Splits this duration into minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (hours: Int, minutes: Int, seconds: Int) -> T): T =
    action(hoursComponent, minutesComponent, secondsComponent)

/**
 * Splits this duration into hours, minutes, seconds, and nanoseconds and executes the given [action] with these components.
 * The result of [action] is returned as the result of this function.
 * `nanoseconds` represents the whole number of nanoseconds in this duration, and its absolute value is less than 1_000_000_000;
 * `seconds` represents the whole number of seconds in this duration, and its absolute value is less than 60;
 * `minutes` represents the whole number of minutes in this duration, and its absolute value is less than 60;
 * `hours` represents the whole number of hours in this duration.
 * If the value doesn't fit in [Int] range, i.e. it's greater than [Int.MAX_VALUE] or less than [Int.MIN_VALUE],
 * it is coerced into that range.
 */
public inline fun <T> toComponents(action: (days: Int, hours: Int, minutes: Int, seconds: Int, nanoseconds: Int) -> T): T =
    action(daysComponent, hoursComponent, minutesComponent, secondsComponent, nanosecondsComponent)

// conversion to units

/**
 * Returns the value of this duration expressed as a [Double] number of the specified [unit].
 */
val inNanoseconds: Int get() = (inNanoseconds % 1e9).toInt()

/**
 * Returns the value of this duration expressed as an [Int] number of the specified [unit].
 */
val inSeconds: Int get() = (inSeconds.toLong()).toInt()

/**
 * Returns the value of this duration expressed as a [Long] number of the specified [unit].
 */
val inMinutes: Int get() = (inMinutes.toLong()).toInt()

/**
 * Returns the value of this duration expressed as a [Long] number of the specified [unit].
 */
val inHours: Int get() = (inHours.toLong()).toInt()

/**
 * Returns the value of this duration expressed as a [Long] number of the specified [unit].
 */
val inDays: Int get() = (inDays.toLong()).toInt()
the range of [Int] type, it is coerced into that range, see the conversion [Double.toInt] for details.

```kotlin
public fun toInt(unit: DurationUnit): Int = toDouble(unit).toInt() /* The value of this duration expressed as a
    [Double] number of seconds. */
    public fun inMinutes: Double get() = toDouble(DurationUnit.MINUTES) /* The value of this duration expressed as a
    [Double] number of milliseconds. */
    public fun inMilliSeconds: Double get() = toDouble(DurationUnit.MILLISECONDS) /* The value of this duration expressed as a
    [Double] number of nanoseconds. */
    public fun inNanoseconds: Double get() = toDouble(DurationUnit.NANOSECONDS) // shortcuts
    override fun toString(): String = when {
        isInfinite() -> value.toString()
        value == 0.0 -> "0s"
        else -> {
            val absNs = absoluteValue.inNanoseconds
            var scientific = false
            var maxDecimals = 0
            val unit = when {
                absNs < 1e-6 -> DurationUnit.SECONDS.also { scientific = true }
                absNs < 1 -> DurationUnit.MICROSECONDS
                absNs < 1e3 -> DurationUnit.MICROSECONDS
                absNs < 1e6 -> DurationUnit.MILLISECONDS
                absNs < 60_000e9 -> DurationUnit.SECONDS
                absNs < 3600_000e9 -> DurationUnit.MINUTES
                absNs < 86400e9 * 1e7 -> DurationUnit.DAYS
                else -> DurationUnit.DAYS.also { scientific = true }
            }
            val value = toDouble(unit)
            when {
                scientific -> formatScientific(value)
                maxDecimals > 0 -> formatUpToDecimals(value, maxDecimals)
                else -> formatToExactDecimals(value, precision(abs(value)))
            } + unit.shortName()
        }
    }
    private fun precision(value: Double): Int = when {
        value < 1 -> 3
        value < 10 -> 2
        value < 100 -> 1
        else -> 0
    } /* * Returns the value of this duration expressed as a [Long] number of nanoseconds.
     * If the value doesn't fit in the range of [Long] type, it is coerced into that range, see the conversion [Double.toLong] for details.
     */
    public fun toLongNanoseconds(): Long = toLong(DurationUnit.NANOSECONDS) /* * Returns the value of this duration expressed as a [Long] number of milliseconds.
     */
    public fun toLongMilliseconds(): Long = toLong(DurationUnit.MILLISECONDS) /* * Returns a string representation of this duration value expressed in the unit which yields the most compact and readable number.
     * Special cases:
     * - zero duration is formatted as "0s"
     * - the infinite duration is formatted as "Infinity"
     * - very small durations (less than 1e-15 s) are expressed in seconds and formatted in scientific notation
     * - very big durations (more than 1e+7 days) are expressed in days and formatted in scientific notation
     * @return the value of duration in the automatically determined unit
     * followed by that unit abbreviated name: d', h', m', s', ms', ms', us', or ns':
     */
    public fun toString(unit: DurationUnit, decimals: Int = 0): String = when {
        absNs < 1e3 -> DurationUnit.NANOSECONDS
        absNs < 1e6 -> DurationUnit.MILLISECONDS
        absNs < 6000_000e9 -> DurationUnit.SECONDS
        absNs < 86400e9 * 1e7 -> DurationUnit.MINUTES
        absNs < 3600_000e9 -> DurationUnit.HOURS
        else -> DurationUnit.DAYS
    }
    /**
     * Returns a string representation of this duration value expressed in the given [unit] and formatted with the specified [decimals] number of
digits after decimal point.
     * Special cases:
     * - the infinite duration is formatted as "Infinity"
     */
    public fun toString(unit: DurationUnit, decimals: Int = 0): String = when {
        abs(number) < 1e14 -> formatToExactDecimals(number, decimals.coerceAtMost(12))
        else -> formatScientific(number)
    }
```

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 2494
* Returns an ISO-8601 based string representation of this duration.

* The returned value is presented in the format `"PT{h}h{m}m{f}s.s"`, where `{h}`, `{m}`, `{s}` are the integer components of this duration (see [toComponents()]).

* and `{f}` is a fractional part of second. Depending on the roundness of the value the fractional part can be formatted with either

* 0, 3, 6, or 9 decimal digits.

* If the hours component absolute value of this duration is greater than [Int.MAX_VALUE], it is replaced with [Int.MAX_VALUE].

* so the infinite duration is formatted as `"PT2147483647H"`.

* Negative durations are indicated with the sign `"-"` in the beginning of the returned string, for example, `"-PT5M30S"`.

* @sample samples.time.Durations.toIsoString

@UseExperimental(ExperimentalStdlibApi::class)
public fun toIsoString(): String = buildString {
    if (isNegative()) append('-')
    append("PT")
    absoluteValue.toComponents { hours, minutes, seconds, nanoseconds ->
        val hasHours = hours != 0
        val hasMinutes = minutes != 0 || (hasHours && hasSeconds)
        if (hasHours) {
            append(hours).append('H')
        }
        if (hasMinutes) {
            append(minutes).append('M')
        }
        if (hasSeconds || (!hasHours && !hasMinutes)) {
            append(seconds)
            if (nanoseconds != 0) {
                append('S')
                when {
                    nanoseconds % 1_000_000 == 0 -> appendRange(nss, 0, 3)
                    nanoseconds % 1_000 == 0 -> appendRange(nss, 0, 6)
                    else -> append(nss)
                }
            }
        }
    }
}

// constructing from number of units
// extension functions

/** Returns a [Duration] equal to this [Int] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Int.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Long] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Long.toDuration(unit: DurationUnit): Duration =
    toDouble().toDuration(unit)

/** Returns a [Duration] equal to this [Double] number of the specified [unit]. */
@SinceKotlin("1.3")
@ExperimentalTime
public fun Double.toDuration(unit: DurationUnit): Duration =
    Duration(convertDurationUnit(this, unit, storageUnit))

// constructing from number of units
// extension properties

/** Returns a [Duration] equal to this [Int] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Long] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Double] number of nanoseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.nanoseconds get() =
    toDuration(DurationUnit.NANOSECONDS)

/** Returns a [Duration] equal to this [Int] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Long] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Double] number of microseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.microseconds get() =
    toDuration(DurationUnit.MICROSECONDS)

/** Returns a [Duration] equal to this [Int] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Long] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Double] number of milliseconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Double.milliseconds get() =
    toDuration(DurationUnit.MILLISECONDS)

/** Returns a [Duration] equal to this [Int] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Int.seconds get() =
    toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Long] number of seconds. */
@SinceKotlin("1.3")
@ExperimentalTime
public val Long.seconds get() =
    toDuration(DurationUnit.SECONDS)
public val Double.seconds get() = toDuration(DurationUnit.SECONDS)

/** Returns a [Duration] equal to this [Int] number of minutes. */
public val Int.minutes get() = toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Long] number of minutes. */
public val Long.minutes get() = toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Double] number of minutes. */
public val Double.minutes get() = toDuration(DurationUnit.MINUTES)

/** Returns a [Duration] equal to this [Int] number of hours. */
public val Int.hours get() = toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Long] number of hours. */
public val Long.hours get() = toDuration(DurationUnit.HOURS)

/** Returns a [Duration] equal to this [Double] number of hours. */
public val Double.hours get() = toDuration(DurationUnit.HOURS)

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
public inline operator fun Int.times(duration: Duration): Duration = duration * this

/** Returns a duration whose value is the specified [duration] value multiplied by this number. */
public inline operator fun Double.times(duration: Duration): Duration = duration * this


internal expect fun formatToExactDecimals(value: Double, decimals: Int): String
internal expect fun formatUpToDecimals(value: Double, decimals: Int): String
internal expect fun formatScientific(value: Double): String

@file:kotlin.jvm.JvmMultifileClass()
@file:kotlin.jvm.JvmName("DurationUnitKt")

package kotlin.time

/**
 * The list of possible time measurement units, in which a duration can be expressed.
 *
 * The smallest time unit is [NANOSECONDS] and the largest is [DAYS], which corresponds to exactly 24 [HOURS].
 */
public expect enum class DurationUnit {
    /**
     * Time unit representing one nanosecond, which is 1/1000 of a microsecond.
     */
    NANOSECONDS,
    /**
     * Time unit representing one microsecond, which is 1/1000 of a millisecond.
     */
    MICROSECONDS,
    /**
     * Time unit representing one millisecond, which is 1/1000 of a second.
     */
    MILISECONDS,
    /**
     * Time unit representing one second.
     */
    SECONDS,
    /**
     * Time unit representing one minute.
     */
    MINUTES,
    /**
     * Time unit representing one hour.
     */
    HOURS,
    /**
     * Time unit representing one day, which is always equal to 24 hours.
     */
    DAYS;
}

/** Converts the given time duration [value] expressed in the specified [sourceUnit] into the specified [targetUnit]. */
public fun Duration.toDurationUnit(value: Double, sourceUnit: DurationUnit?, targetUnit: DurationUnit?): Duration = when (sourceUnit) {
    DurationUnit.NANOSECONDS -> value
    DurationUnit.MICROSECONDS -> value * 1000
    DurationUnit.MILISECONDS -> value * 1000
    DurationUnit.SECONDS -> value * 60
    DurationUnit.MINUTES -> value * 60
    DurationUnit.HOURS -> value * 24
    DurationUnit.DAYS -> value / 1000
}
This annotation marks the experimental preview of the standard library API for measuring time and working with durations. Note that this API is in a preview state and has a very high chance of being changed in the future. Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library. Any usage of a declaration annotated with `@ExperimentalTime` must be accepted either by annotating that usage with the [OptIn] annotation, e.g. @OptIn(ExperimentalTime::class), or by using the compiler argument `-X-opt-in=kotlin.time.ExperimentalTime`

```kotlin
/**
 * This annotation marks the experimental preview of the standard library API for measuring time and working with durations.
 *
 * Note that this API is in a preview state and has a very high chance of being changed in the future.
 * Do not use it if you develop a library since your library will become binary incompatible with the future versions of the standard library. Any usage of a declaration annotated with `@ExperimentalTime` must be accepted either by annotating that usage with the [OptIn] annotation, e.g. @OptIn(ExperimentalTime::class), or by using the compiler argument `-X-opt-in=kotlin.time.ExperimentalTime`
 */
@Suppress("DEPRECATION")
@Experimental(level = Experimental.Level.ERROR)
@RequiresOptIn(level = RequiresOptIn.Level.ERROR)
@MustBeDocumented
@Retention(AnnotationRetention.BINARY)
@Target(
    CLASS,
    ANNOTATION_CLASS,
    PROPERTY,
    FIELD,
    LOCAL_VARIABLE,
    VALUE_PARAMETER,
    CONSTRUCTOR,
    FUNCTION,
    PROPERTY_GETTER,
    PROPERTY_SETTER,
    TYPEALIAS
)
@SinceKotlin("1.3")
public annotation class ExperimentalTime

/**
 * Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the LICENSE.txt file.
 */

package kotlin.time

/**
 * A source of time for measuring time intervals.
 * The only operation provided by the time source is [markNow]. It returns a [TimeMark], which can be used to query the elapsed time later.
 * Note that this API is in a preview state and has a very high chance of being changed in the future.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public interface TimeSource {
    /**
     * Marks a point in time on this time source.
     * The returned [TimeMark] instance encapsulates the captured time point and allows querying the duration of time interval [elapsed] from that point.
     */
    public fun markNow(): TimeMark

    /**
     * The most precise time source available in the platform.
     * This time source returns its readings from a source of monotonic time when it is available in a target platform, and resorts to a non-monotonic time source otherwise.
     */
    public object Monotonic : TimeSource by MonotonicTimeSource {
        override fun toString(): String = MonotonicTimeSource.toString()
    }

    companion object {
    }
}

/**
 * Represents a time point notched on a particular [TimeSource]. Remains bound to the time source it was taken from and allows querying for the duration of time elapsed from that point (see the function [elapsedNow]).
 */
@SinceKotlin("1.3")
@ExperimentalTime
public abstract class TimeMark {
    /**
     * Returns the amount of time passed from this mark measured with the time source from which this mark was taken.
     */
    public abstract fun markNow(): TimeMark

    /**
     * Returns a time mark on the same time source that is ahead of this time mark by the specified [duration].
     */
    public open operator fun plus(duration: Duration): TimeMark = AdjustedTimeMark(this, duration)

    /**
     * Returns a time mark on the same time source that is behind this time mark by the specified [duration].
     */
    public open operator fun minus(duration: Duration): TimeMark = plus(-duration)

    /**
     * Returns true if this time mark has passed according to the time source from which this mark was taken.
     */
    public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()

    /**
     * Returns false if this time mark has not passed according to the time source from which this mark was taken.
     */
    public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()

    /**
     * Returns true if this time mark has passed according to the time source from which this mark was taken.
     */
    public fun hasPassedNow(): Boolean = !elapsedNow().isNegative()

    /**
     * Returns false if this time mark has not passed according to the time source from which this mark was taken.
     */
    public fun hasNotPassedNow(): Boolean = elapsedNow().isNegative()
Comparing one TimeMark to another is not a well defined operation because these time marks could have been obtained from the different time sources.  

override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

private class AdjustedTimeMark(val mark: TimeMark, val adjustment: Duration) : TimeMark() {
    override fun elapsedNow(): Duration = mark.elapsedNow() - adjustment

    override fun plus(duration: Duration): TimeMark = AdjustedTimeMark(mark, adjustment + duration)
}

private class LongTimeMark(private val startedAt: Long, private val timeSource: AbstractLongTimeSource, private val offset: Duration) : TimeMark() {
        override fun elapsedNow(): Duration = (timeSource.read() - startedAt).toDuration(timeSource.unit) - offset
        override fun plus(duration: Duration): TimeMark = LongTimeMark(startedAt, timeSource, offset + duration)
    }

override fun markNow(): TimeMark = LongTimeMark(read(), this, Duration.ZERO)
}

/**
 * A time source that has programmatically updatable readings. It is useful as a predictable source of time in tests.
 *
 * The current reading value can be advanced by the specified duration amount with the operator 
 * `plusAssign`:  
 * ```
 * val timeSource = TestTimeSource()
 * timeSource += 10.seconds  
 * ```
 *
 * Implementation note: the current reading value is stored as a [Long] number of nanoseconds, thus it's capable to represent a time range of approximately ±2^62 years.
 * Should the reading value overflow as the result of [plusAssign] operation, an [IllegalStateException] is thrown.
 *
 * Advance the current reading value of this time source by the specified [duration].
 *
 * For example, if the duration being added is `0.6.nanoseconds`, the reading doesn't advance because the duration value is rounded to zero nanoseconds.
 *
 * Throws [IllegalStateException] when the reading value overflows as the result of this operation.
 */
@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeSource interface instead.", ReplaceWith("TimeSource", "kotlin.time.TimeSource"))
public typealias Clock = TimeSource

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeMark class instead.", ReplaceWith("TimeMark", "kotlin.time.TimeMark"))
public typealias ClockMark = TimeMark

fits in long, add it as long
val newReading = reading + longDelta
else  
    // when delta is
greater than long, add it as double
val newReading = reading + delta

Long.MAX_VALUE || newReading < Long.MIN_VALUE) overflow(duration) in newReading.toDouble() 

private fun overflow(duration: Duration) in throw IllegalArgumentException("TestTimeSource will
overflow if its reading $reading ns is advanced by $duration.")

}

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TimeSource.Monotonic instead.",
ReplaceWith("TimeSource.Monotonic", "kotlin.time.TimeSource"))

public typealias MonoClock =
TimeSource.Monotonic

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use AbstractLongTimeSource instead.",
ReplaceWith("AbstractLongTimeSource", "kotlin.time.AbstractLongTimeSource")

public typealias AbstractLongClock =
AbstractLongTimeSource

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use AbstractDoubleTimeSource instead.",
ReplaceWith("AbstractDoubleTimeSource", "kotlin.time.AbstractDoubleTimeSource")

public typealias AbstractDoubleClock =
AbstractDoubleTimeSource

@SinceKotlin("1.3")
@ExperimentalTime
@Deprecated("Use TestTimeSource
instead.", ReplaceWith("TestTimeSource", "kotlin.time.TestTimeSource")

public typealias TestClock =
TestTimeSource

package kotlin.time

import kotlin.contracts.*

/**
 * Executes the given function [block] and returns the
 * duration of elapsed time interval.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun measureTime(block: () -> Unit): Duration {

contract {
    callsInPlace(block, InvocationKind.EXACTLY_ONCE)
}

return
TimeSource.measureTime(block)

/**
 * Executes the given function [block] and returns the
 * duration of elapsed time interval.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun TimeSource.measureTime(block: () -> Unit): Duration {

contract {
    callsInPlace(block, InvocationKind.EXACTLY_ONCE)
}

val mark = markNow()
block()
return mark.elapsedNow()

val

/**
 * Data class representing a result of
 * executing an action, along with the duration of elapsed time interval.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public data class TimedValue<T>(val value: T, val duration: Duration)

/**
 * Executes the given function [block] and returns an instance of [TimedValue] class, containing
 * both the result of the function execution and the duration of elapsed time interval.
 */
@SinceKotlin("1.3")
@ExperimentalTime
public inline fun <T> TimeSource.measureTimedValue(block: () -> T): TimedValue<T> {

contract {
    callsInPlace(block, InvocationKind.EXACTLY_ONCE)
}

val mark = markNow()
val result = block()
return TimedValue(result, mark.elapsedNow())

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */

package kotlin

import kotlin.internal.InlineOnly

/**
 * Returns a hash code value for the object or zero if the object
 * is `null`.
 */
@SinceKotlin("1.3")
@InlineOnly
public inline fun Any?.hashCode(): Int = this?.hashCode() ?: 0

/*
 * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
 */
library.\n *\n * [major], [minor] and [patch] are integer components of a version,\n * they must be non-negative and
not greater than 255 ([MAX_COMPONENT_VALUE]).\n *\n * @constructor Creates a version from all three
components.\n */\n@SinceKotlin(\"1.1\")\npublic class KotlinVersion(val major: Int, val minor: Int, val patch: Int) :
Comparable<KotlinVersion> {\n /**\n * Creates a version from [major] and [minor] components, leaving
[patch] component zero.\n */\n public constructor(major: Int, minor: Int) : this(major, minor, 0)\n\n private val
version = versionOf(major, minor, patch)\n\n private fun versionOf(major: Int, minor: Int, patch: Int): Int {\n
require(major in 0..MAX_COMPONENT_VALUE && minor in 0..MAX_COMPONENT_VALUE && patch in
0..MAX_COMPONENT_VALUE) {\n
\"Version components are out of range: $major.$minor.$patch\"\n
}\n
return major.shl(16) + minor.shl(8) + patch\n }\n\n /**\n * Returns the string representation of this
version\n */\n override fun toString(): String = \"$major.$minor.$patch\"\n\n override fun equals(other:
Any?): Boolean {\n
if (this === other) return true\n
val otherVersion = (other as? KotlinVersion) ?: return
false\n
return this.version == otherVersion.version\n }\n\n override fun hashCode(): Int = version\n\n
override fun compareTo(other: KotlinVersion): Int = version - other.version\n\n /**\n * Returns `true` if this
version is not less than the version specified\n * with the provided [major] and [minor] components.\n */\n
public fun isAtLeast(major: Int, minor: Int): Boolean = // this.version >= versionOf(major, minor, 0)\n
this.major > major || (this.major == major &&\n
this.minor >= minor)\n\n /**\n * Returns `true` if this
version is not less than the version specified\n * with the provided [major], [minor] and [patch] components.\n
*/\n public fun isAtLeast(major: Int, minor: Int, patch: Int): Boolean = // this.version >= versionOf(major, minor,
patch)\n
this.major > major || (this.major == major &&\n
(this.minor > minor || this.minor == minor
&&\n
this.patch >= patch))\n\n companion object {\n
/**\n
* Maximum value a version
component can have, a constant value 255.\n
*/\n
// NOTE: Must be placed before CURRENT because its
initialization requires this field being initialized in JS\n
public const val MAX_COMPONENT_VALUE =
255\n\n
/**\n
* Returns the current version of the Kotlin standard library.\n
*/\n
@kotlin.jvm.JvmField\n
public val CURRENT: KotlinVersion = KotlinVersion(1, 3, 70) // value is written here
automatically during build\n }\n}\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming
Language contributors.\n * Use of this source code is governed by the Apache 2.0 license that can be found in the
license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmName(\"LateinitKt\")\n@file:Suppress(\"unused\")\n\npackage kotlin\n\nimport
kotlin.internal.InlineOnly\nimport kotlin.internal.AccessibleLateinitPropertyLiteral\nimport
kotlin.reflect.KProperty0\n\n/**\n * Returns `true` if this lateinit property has been assigned a value, and `false`
otherwise.\n *\n * Cannot be used in an inline function, to avoid binary compatibility issues.\n
*/\n@SinceKotlin(\"1.2\")\n@InlineOnly\ninline val @receiver:AccessibleLateinitPropertyLiteral
KProperty0<*>.isInitialized: Boolean\n get() = throw NotImplementedError(\"Implementation is
intrinsic\")\n","/*\n * Copyright 2010-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.\n * Use
of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.\n
*/\n\n@file:kotlin.jvm.JvmName(\"LazyKt\")\n@file:kotlin.jvm.JvmMultifileClass\n\npackage kotlin\n\nimport
kotlin.reflect.KProperty\n\n/**\n * Represents a value with lazy initialization.\n *\n * To create an instance of
[Lazy] use the [lazy] function.\n */\npublic interface Lazy<out T> {\n /**\n * Gets the lazily initialized value of
the current Lazy instance.\n * Once the value was initialized it must not change during the rest of lifetime of this
Lazy instance.\n */\n public val value: T\n\n /**\n * Returns `true` if a value for this Lazy instance has been
already initialized, and `false` otherwise.\n * Once this function has returned `true` it stays `true` for the rest of
lifetime of this Lazy instance.\n */\n public fun isInitialized(): Boolean\n}\n\n/**\n * Creates a new instance of
the [Lazy] that is already initialized with the specified [value].\n */\npublic fun <T> lazyOf(value: T): Lazy<T> =
InitializedLazyImpl(value)\n\n/**\n * An extension to delegate a read-only property of type [T] to an instance of
[Lazy].\n *\n * This extension allows to use instances of Lazy for property delegation:\n * `val property: String by
lazy { initializer }`\n */\n@kotlin.internal.InlineOnly\npublic inline operator fun <T> Lazy<T>.getValue(thisRef:
Any?, property: KProperty<*>): T = value\n\n/**\n * Specifies how a [Lazy] instance synchronizes initialization
among multiple threads.\n */\npublic enum class LazyThreadSafetyMode {\n\n /**\n * Locks are used to ensure

Open Source Used In IOx-FogDirector-AC18 1.17.0 2500


that only a single thread can initialize the [Lazy] instance. *

SYNCHRONIZED;

*/

* Initialize function can be called several times on concurrent access to uninitialized [Lazy] instance value; *

* but only the first returned value will be used as the value of [Lazy] instance value; if the instance is accessed from multiple
threads, its behavior is undefined. *

* This mode should not be used unless the [Lazy] instance is

* guaranteed never to be initialized from more than one thread.*

* NONE, */

UNINITIALIZED_VALUE

// internal to be called from lazy in JS

internal class UnsafeLazyImpl<out

T>(initializer: () -> T): Lazy<T>, Serializable {

    private var initializer: (() -> T)? = initializer

    private var

    _value: Any? = UNINITIALIZED_VALUE

    override val value: T
        get() {
            if (_value ===

            UNINITIALIZED_VALUE) {
                _value = initializer!!()
                initializer = null
            }

            return _value as T
        }

    override fun isInitialized():

        return _value !== UNINITIALIZED_VALUE

    override fun toString(): String = if (isInitialized())

        value.toString() else "Lazy value not initialized yet."

    private fun writeReplace(): Any =

        InitializedLazyImpl(value)

}

internal class InitializedLazyImpl<out T>(override val value: T) : Lazy<T>, Serializable {

    override fun isInitialized(): Boolean = true

    override fun toString(): String =

        value.toString()

}

@file:kotlin.jvm.JvmMultifileClass

@file:kotlin.jvm.JvmName("NumbersKt")

package kotlin

/**
 * Counts the number of set bits in the binary representation of this [Int] number.
 */

@SinceKotlin("1.3")

@ExperimentalStdlibApi

public expect fun Int.countOneBits(): Int

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Int] number.
 */

@SinceKotlin("1.3")

@ExperimentalStdlibApi

public expect fun Int.countLeadingZeroBits(): Int

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int]

* number.

* Counts the number of consecutive least significant bits that are zero in the binary representation of this [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.

* Counts the number of consecutive least significant bits in the integer representation of [Int]

* number.
Long.countTrailingZeroBits(): Int

int n /** Returns a number having a single bit set in the position of the most significant set bit of this [Long] number, \n * or zero, if this number is zero. */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Long.takeHighestBit(): Long

int n /** Returns a number having a single bit set in the position of the least significant set bit of this [Long] number, \n * or zero, if this number is zero. */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
public fun Long.takeLowestBit(): Long

Long.rotateLeft(bitCount: Int): Long

Long.rotateRight(bitCount: Int): Long

Byte.countOneBits(): Int = (toInt() and 0xFF).countOneBits()

Byte.countLeadingZeroBits(): Int = (toInt() and 0xFF).countLeadingZeroBits() - (Int.SIZE_BITS - Byte.SIZE_BITS)

Byte.countTrailingZeroBits(): Int = (toInt() or 0x100).countTrailingZeroBits()

Byte.takeHighestBit(): Byte = (toInt() and 0xFF).takeHighestBit().toByte()

Byte.takeLowestBit(): Byte = toInt().takeLowestBit().toByte()

Byte.rotateLeft(bitCount: Int): Byte =
(toint().shl(bitCount and 7) or (toInt() and 0xFF).ushr(8 - (bitCount and 7))).toByte()

Byte.rotateRight(bitCount: Int): Byte =
(toint().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()

Byte.rotateRight(bitCount: Int): Byte =
(toint().shl(8 - (bitCount and 7)) or (toInt() and 0xFF).ushr(bitCount and 7)).toByte()
public inline fun Short.countOneBits(): Int = (toInt() and 0xFFFF).countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [Short] number.
 */
public inline fun Short.countLeadingZeroBits(): Int = (toInt() and 0xFFFF).countLeadingZeroBits() - (Int.SIZE_BITS - Short.SIZE_BITS)

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [Short] number.
 */
public inline fun Short.countTrailingZeroBits(): Int = (toInt() or 0x10000).countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [Short] number.
 */
public inline fun Short.takeHighestOneBit(): Short = (toInt() and 0xFFFF).takeHighestOneBit().toShort()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [Short] number.
 */
public inline fun Short.takeLowestOneBit(): Short = toInt().takeLowestOneBit().toShort()

/**
 * Rotates the binary representation of this [Short] number left by the specified [bitCount] number of bits.
 */
public fun Short.rotateLeft(bitCount: Int): Short =
(toInt().shl(bitCount and 15) or (toInt() and 0xFFFF).ushr(16 - (bitCount and 15))).toShort()

/**
 * Rotates the binary representation of this [Short] number right by the specified [bitCount] number of bits.
 */
public fun Short.rotateRight(bitCount: Int): Short =
(toInt().shl(16 - (bitCount and 15)) or (toInt() and 0xFFFF).ushr(bitCount and 15)).toShort()
A.to(that: B): Pair<A, B> = Pair(this, that)

/**
 * Converts this pair into a list.
 * @sample
code samples/misc.Tuples.pairToList
 */
public fun <T> Pair<T, T>.toList(): List<T> = listOf(first, second)

/**
 * Represents a triad of values
 *
 * There is no meaning attached to values in this class, it can be used for any purpose.
 * Triple exhibits value semantics, i.e. two triples are equal if all three components are equal.
 * An example of decomposing it into values:
 */
public data class Triple<out A, out B, out C>(
    public val first: A,
    public val second: B,
    public val third: C
) : Serializable

    /**
     * Returns string representation of the [Triple] including its [first], [second] and [third] values.
     */
    public override fun toString(): String = "($first, $second, $third)"

    companion object {
        /**
         * An empty range of values of type UInt.
         */
        public val EMPTY: UIntRange = UIntRange(UInt.MAX_VALUE, UInt.MIN_VALUE)
    }

    /**
     * A progression of values of type `UInt`.
     */
    @SinceKotlin("1.3")
    @ExperimentalUnsignedTypes
    public open class UIntProgression
    internal constructor(
        start: UInt,
        endInclusive: UInt,
        step: Int
    ) : Iterable<UInt>

    /**
     * The first element in the progression.
     */
    public val first: UInt = start

    /**
     * The last element in the progression.
     */
    public val last: UInt = getProgressionLastElement(start, endInclusive, step)

    /**
     * The step of the progression.
     */
    public val step: Int = step

    override fun iterator(): UIntIterator = UIntProgressionIterator(first, last, step)

    /**
     * Checks if the progression is empty.
     */
    public fun isEmpty(): Boolean = if (step > 0) first > last else first < last

    override fun equals(other: Any?): Boolean =
        other is UIntProgression && (isEmpty() && other.isEmpty() ||
                first == other.first && last == other.last && step == other.step)

    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * first.toInt() + last.toInt() + step.toInt())

    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"

    companion object {
        /**
         * Creates UIntProgression within the specified bounds of a closed range.
         */
        public fun fromClosedRange(rangeStart: UInt, rangeEnd: UInt, step: Int): UIntProgression = UIntProgression(rangeStart, rangeEnd, step)
    }

    private class UIntProgressionIterator(first: UInt, last: UInt, step: Int) : UIntIterator() {
        private val finalElement = last
        private var hasNext: Boolean = if (step > 0) first <= last else first >= last
        private val step = step.toUInt() // use 2-complement math for negative steps

        override fun hasNext(): Boolean = hasNext

        override fun nextUInt(): UInt {
            val value = next
            if (value == finalElement) {
                if (!hasNext) throw kotlin.NoSuchElementException()
                hasNext = false
            }
else {
    next += step
    return value
}

} /* Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors. * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

public class IOxFogDirectorAC181.17.0 {
    public static final int 2505
    {
        @Override
        public abstract class UByteIterator : Iterator<UByte> {
            override final fun next() = nextUByte()
        }
        return value
    }
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

// Auto-generated file. DO NOT EDIT!

package kotlin.collections

/** An iterator over a sequence of values of type `UByte`. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UByteIterator : Iterator<UByte> {
    override final fun next() = nextUByte()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUByte(): UByte
}

/** An iterator over a sequence of values of type `UShort`. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UShortIterator : Iterator<UShort> {
    override final fun next() = nextUShort()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUShort(): UShort
}

/** An iterator over a sequence of values of type `UInt`. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class UIntIterator : Iterator<UInt> {
    override final fun next() = nextUInt()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextUInt(): UInt
}

/** An iterator over a sequence of values of type `ULong`. */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public abstract class ULongIterator : Iterator<ULong> {
    override final fun next() = nextULong()

    /** Returns the next value in the sequence without boxing. */
    public abstract fun nextULong(): ULong
}

/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file. */

// Auto-generated file. DO NOT EDIT!

package kotlin.ranges

import kotlin.internal.*

/**
 * A range of values of type `ULong`.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongRange(start: ULong, endInclusive: ULong) : ULongProgression(start, endInclusive, 1), ClosedRange<ULong> {
    override val start: ULong get() = first
    override val endInclusive: ULong get() = last
    override fun contains(value: ULong): Boolean = first <= value && value <= last
    override fun isEmpty(): Boolean = first > last
    override fun equals(other: Any?): Boolean =
        other is ULongRange && (isEmpty() && other.isEmpty() ||
                first == other.first && last == other.last)
    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt()) + (step xor (step ushr 32)).toInt())
    override fun toString(): String = if (step > 0) "$first..$last step $step" else "$first downTo $last step ${-step}"
}

/**
 * A progression of values of type `ULong`. *
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public class ULongProgression
internal constructor(
    start: ULong,
    endInclusive: ULong,
    step: Long
) : Iterable<ULong> {
    init {
        if (step == 0.toLong()) throw kotlin.IllegalArgumentException("Step must be non-zero.")
        if (step == Long.MIN_VALUE) throw kotlin.IllegalArgumentException("Step must be greater than Long.MIN_VALUE to avoid overflow on negation.")
    }
    override fun iterator(): ULongIterator =
        ULongProgressionIterator(first, last, step)
    /** Checks if the progression is empty. */
    public var val empty: Boolean = false
    override fun isEmpty(): Boolean =
        first == last && step == 0
    override fun equals(other: Any?): Boolean =
        other is ULongProgression && (isNotEmpty() && other.isNotEmpty())
    /** The first element in the progression. */
    public var val first: ULong = start
    /** The last element in the progression. */
    public var val last: ULong = (first + step - 1)
    /** The step of the progression. */
    public var val step: Long = step
    override fun hashCode(): Int =
        if (isEmpty()) -1 else (31 * (first xor (first shr 32)).toInt() + (last xor (last shr 32)).toInt())
    override fun toString(): String = "$first..$last step $step"
the value is incremented on each step.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
private class

ULongProgressionIterator(first: ULong, last: ULong, step: Long) : ULongIterator() {
    private val finalElement = last
    private var hasNext: Boolean = if (step > 0) first <= last else first >= last
    private var value = first
    private var step = step.toULong() // use 2-complement math for negative steps
    private var next = if (hasNext) first else
    
    override fun hasNext(): Boolean = hasNext
    
    override fun nextULong(): ULong {
        val
        if (value == finalElement) {
            if (!hasNext) throw kotlin.NoSuchElementException()
            hasNext = false
        } else {
            next += step
        }
        return value
    }
}

"/*
 * Copyright
 * 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */

package kotlin.math

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: UInt, b: UInt): UInt {
    return minOf(a, b)
}

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun min(a: ULong, b: ULong): ULong {
    return minOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: UInt, b: UInt): UInt {
    return maxOf(a, b)
}

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun max(a: ULong, b: ULong): ULong {
    return maxOf(a, b)
}

"/*
 * Copyright 2010-2019 JetBrains s.r.o. and Kotlin Programming Language contributors.
 * Use of this source code is governed by the Apache 2.0 license that can be found in
 * the license/LICENSE.txt file.
 */

@file:kotlin.jvm.JvmName("UNumbersKt")
package

/**
 * Counts the number of set bits in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countOneBits(): Int = toInt().countOneBits()

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countLeadingZeroBits(): Int = toInt().countLeadingZeroBits()

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UInt] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.countTrailingZeroBits(): Int = toInt().countTrailingZeroBits()

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UInt] number, or zero, if this number
 * is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeHighestOneBit(): UInt = toInt().takeHighestOneBit().toUInt()

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [UInt] number, or zero, if this number
 * is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.takeLowestOneBit(): UInt = toInt().takeLowestOneBit().toUInt()

/**
 * Rotates the binary representation of this [UInt] number left by the specified [bitCount] number of bits.
 * Most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count.
 *
 * number.rotateLeft(-n) == number.rotateRight(n) * * * Rotating by a multiple of [UInt.SIZE_BITS] (32) returns the same number, or more generally
 * number.rotateLeft(n) == number.rotateLeft(n % 32)
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateLeft(bitCount: Int): UInt = toInt().rotateLeft(bitCount).toUInt()

/**
 * Rotates the binary representation of this [UInt] number right by the specified [bitCount] number of bits.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnly
public inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt()
the same number, or more generally
`number.rotateRight(n) == number.rotateRight(n % 32)`

`@SinceKotlin("1.3")` @ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UInt.rotateRight(bitCount: Int): UInt = toInt().rotateRight(bitCount).toUInt()


/**
 * Counts the number of set bits in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countOneBits(): Int = toLong().countOneBits().toInt()/* */


/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countLeadingZeroBits(): Int = toLong().countLeadingZeroBits().toInt()/* */


/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [ULong] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.countTrailingZeroBits(): Int = toLong().countTrailingZeroBits().toInt()/* */

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [ULong] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.takeHighestOneBit(): ULong = toLong().takeHighestOneBit().toULong()/* */

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [ULong] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.takeLowestOneBit(): ULong = toLong().takeLowestOneBit().toULong()/* */

/**
 * Rotates the binary representation of this [ULong] number left by the specified [bitCount] number of bits.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:
 * `number.rotateLeft(-n) == number.rotateRight(n)`
 * Rotating by a multiple of [ULong.SIZE_BITS] returns the same number, or more generally
 * `number.rotateLeft(n) == number.rotateLeft(n % 64)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.rotateLeft(bitCount: Int): ULong = toLong().rotateLeft(bitCount).toULong()/* */

/**
 * Rotates the binary representation of this [ULong] number right by the specified [bitCount] number of bits.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
 * `number.rotateRight(-n) == number.rotateLeft(n)`
 * Rotating by a multiple of [ULong.SIZE_BITS] returns the same number, or more generally
 * `number.rotateRight(n) == number.rotateRight(n % 64)`
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun ULong.rotateRight(bitCount: Int): ULong = toLong().rotateRight(bitCount).toULong()/* */

/**
 * Counts the number of set bits in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UByte.countOneBits(): Int = toUInt().countOneBits()/* */

/**
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UByte.countLeadingZeroBits(): Int = toByte().countLeadingZeroBits()/* */

/**
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UByte] number.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UByte.countTrailingZeroBits(): Int = toByte().countTrailingZeroBits()/* */

/**
 * Returns a number having a single bit set in the position of the most significant set bit of this [UByte] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UByte.takeHighestOneBit(): UByte = toInt().takeHighestOneBit().toUByte()/* */

/**
 * Returns a number having a single bit set in the position of the least significant set bit of this [UByte] number, or zero, if this number is zero.
 */
@SinceKotlin("1.3")
@ExperimentalStdlibApi
@ExperimentalUnsignedTypes
@kotlin.internal.InlineOnl
y

public inline fun UByte.takeLowestOneBit(): UByte = toInt().takeLowestOneBit().toUByte()/* */
this number is zero.

```kotlin
/*
 * Rotates the binary representation of this [UByte] number left by the specified [bitCount] number of bits.
 * The most significant bits pushed out from the left side reenter the number as the least significant bits on the right side.
 * number.rotateLeft(-n) == number.rotateRight(n)
 */
public inline fun UByte.rotateLeft(bitCount: Int): UByte = toByte().rotateLeft(bitCount).toUByte()

/*
 * Rotates the binary representation of this [UByte] number right by the specified [bitCount] number of bits.
 * The least significant bits pushed out from the right side reenter the number as the most significant bits on the left side.
 * number.rotateRight(-n) == number.rotateLeft(n)
 */
public inline fun UByte.rotateRight(bitCount: Int): UByte = toByte().rotateRight(bitCount).toUByte()

/*
 * Counts the number of set bits in the binary representation of this [UShort] number.
 */
public inline fun UShort.countOneBits(): Int = toUInt().countOneBits()

/*
 * Counts the number of consecutive most significant bits that are zero in the binary representation of this [UShort] number.
 */
public inline fun UShort.countLeadingZeroBits(): Int = toShort().countLeadingZeroBits()

/*
 * Counts the number of consecutive least significant bits that are zero in the binary representation of this [UShort] number.
 */
public inline fun UShort.countTrailingZeroBits(): Int = toShort().countTrailingZeroBits()

/*
 * Returns a number having a single bit set in the position of the most significant set bit of this [UShort] number, or zero, if this number is zero.
 */
public inline fun UShort.takeHighestOneBit(): UShort = toInt().takeHighestOneBit().toUShort()

/*
 * Returns a number having a single bit set in the position of the least significant set bit of this [UShort] number, or zero, if this number is zero.
 */
public inline fun UShort.takeLowestOneBit(): UShort = toInt().takeLowestOneBit().toUShort()

/*
 * Rotates the binary representation of this [UShort] number left by the specified [bitCount] number of bits.
 * Rotating the number left by a negative bit count is the same as rotating it right by the negated bit count:
 * number.rotateLeft(-n) == number.rotateRight(n)
 */
public inline fun UShort.rotateLeft(bitCount: Int): UShort = toShort().rotateLeft(bitCount).toUShort()

/*
 * Rotates the binary representation of this [UShort] number right by the specified [bitCount] number of bits.
 * Rotating the number right by a negative bit count is the same as rotating it left by the negated bit count:
 * number.rotateRight(-n) == number.rotateLeft(n)
 */
public inline fun UShort.rotateRight(bitCount: Int): UShort = toShort().rotateRight(bitCount).toUShort()
```
private fun differenceModulo(a: UInt, b: UInt, c: UInt): UInt {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

private fun differenceModulo(a: ULong, b: ULong, c: ULong): ULong {
    val ac = a % c
    val bc = b % c
    return if (ac >= bc) ac - bc else ac - bc + c
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range \([start] to [end]\) in case of a positive \([step]\), or from \([end]\) to \([start]\) in case of a negative \([step]\).
 *
 * No validation on passed parameters is performed. The given parameters should satisfy the condition: \(\*n\) - either \`step > 0` and \`start <= end\`, \*n\) - or \`step < 0` and \`start >= end\.
 *
 * @param start first element of the progression
 * @param step increment, or difference of successive elements in the progression
 * @param end ending bound for the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
@SinceKotlin("1.3")
internal fun getProgressionLastElement(start: UInt, end: UInt, step: Int): UInt = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toUInt())
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toUInt())
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Calculates the final element of a bounded arithmetic progression, i.e. the last element of the progression which is in the range \([start] to [end]\) in case of a positive \([step]\), or from \([end]\) to \([start]\) in case of a negative \([step]\).
 *
 * No validation on passed parameters is performed. The given parameters should satisfy the condition: \(\*n\) - either \`step > 0` and \`start <= end\`, \*n\) - or \`step < 0` and \`start >= end\.
 *
 * @param start first element of the progression
 * @param step increment, or difference of successive elements in the progression
 * @param end ending bound for the progression
 * @return the final element of the progression
 * @suppress
 */
@PublishedApi
@SinceKotlin("1.3")
internal fun getProgressionLastElement(start: ULong, end: ULong, step: Long): ULong = when {
    step > 0 -> if (start >= end) end else end - differenceModulo(end, start, step.toULong())
    step < 0 -> if (start <= end) end else end + differenceModulo(start, end, (-step).toULong())
    else -> throw kotlin.IllegalArgumentException("Step is zero.")
}

/**
 * Returns a string representation of this \([Byte]\) value in the specified \([radix]\). If \([radix]\) is not a valid radix for number to string conversion throws \(\text{Illegalargumentexception}\).
 *
 * @return \(\text{string representation of unsigned number}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this \([Short]\) value in the specified \([radix]\). If \([radix]\) is not a valid radix for number to string conversion throws \(\text{Illegalargumentexception}\).
 *
 * @return \(\text{string representation of unsigned number}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.toString(radix: Int): String = this.toInt().toString(radix)

/**
 * Returns a string representation of this \([Int]\) value in the specified \([radix]\). If \([radix]\) is not a valid radix for number to string conversion throws \(\text{Illegalargumentexception}\).
 *
 * @return \(\text{string representation of unsigned number}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.toString(radix: Int): String = this.toLong().toString(radix)

/**
 * Parses the string as a signed \([UByte]\) number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @return \(\text{signed UByte}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UByte.toUByteOrNull(): UByte = numberFormatError(this)

/**
 * Parses the string as a signed \([UShort]\) number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @return \(\text{signed UShort}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UShort.toUShort(): UShort = numberFormatError(this)

/**
 * Parses the string as a signed \([UInt]\) number and returns the result.
 * @throws NumberFormatException if the string is not a valid representation of a number.
 * @return \(\text{signed UInt}\)
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun UInt.toUInt(): UInt = numberFormatError(this)
representation of a number.

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @throws NumberFormatException if the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws IllegalArgumentException when [radix] is not a valid radix for string to number conversion.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.

* @throws NumberFormatException when the string is not a valid representation of a number.
(digit < 0) return null
     if (result > limitBeforeMul) { return null
            if (limitBeforeMul == limitForMaxRadix) { limitBeforeMul = limit / uradix
            if (result > limitBeforeMul) { return null
    }
            else { return null
        }
        result *= uradix

        val beforeAdding = result
        result += digit.toUInt()
        if (result < beforeAdding) return null // overflow has happened
    }

    return result
}

/**
 * Parses the string as an [ULong] number and returns the result
 * or `null` if the string is not a valid representation of a number.
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun

/**
 * Parses the string as an [ULong] number
 * and returns the result
 * or `null` if the string is not a valid representation of a number.
 *
 * @throws
 */
@SinceKotlin("1.3")
@ExperimentalUnsignedTypes
public fun String.toULongOrNull(radix: Int): ULong? {
    checkRadix(radix)
    val length = this.length
    if (length == 0) return null
    val limit: ULong = ULong.MAX_VALUE
    val start: Int

    val firstChar = this[0]
    if (firstChar < '0') {
        if (length == 1 || firstChar != '+') return null
        start = 1
    } else {
        start = 0
    }

    val limitForMaxRadix = 512409557603043100uL // limit / 36
    var limitBeforeMul = limitForMaxRadix
    val uradix = radix.toULong()
    var result = 0uL
    for (i in start until length) {
        val digit = digitOf(this[i], radix)

        if (digit < 0) return null
        if (result > limitBeforeMul) {
            if (limitBeforeMul == limitForMaxRadix) {
                limitBeforeMul = limit / uradix
                if (result > limitBeforeMul) {
                    return null
                }
            } else {
                return null
            }
        }

        result *= uradix

        val beforeAdding = result
        result += digit.toUInt()
        if (result < beforeAdding) return null // overflow has happened
    }

    return result
}
ln(x: Double): Double

/**
 * Computes the common logarithm (base 10) of the value [x].
 *
 * @see [ln]
 *   - `ln(+Inf)` is `+Inf`
 *   - `ln(0.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun log(x: Double, base: Double): Double

/**
 * Computes the natural logarithm (base `E`) of the value [x].
 *
 * Special cases:
 *   - `ln(NaN)` is `NaN`
 *   - `ln(x)` is `NaN` when `x < 0.0`
 *   - `ln(0.0)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun ln(x: Double): Double

/**
 * Computes `exp(x) - 1`.
 *
 * See also logarithm functions for common fixed bases: [ln], [log10] and [log2].
 */
@SinceKotlin("1.2")
public expect fun expm1(x: Double): Double

/**
 * Computes `exp(x)`. The returned value is `+Inf` for `x > +Inf` and `-Inf` for `x < -Inf`
 */
@SinceKotlin("1.2")
public expect fun exp(x: Double): Double

/**
 * Computes `sqrt(x^2)`. The returned value is `NaN` when `x < 0` or `x` is `NaN`
 */
@SinceKotlin("1.2")
public expect fun sqrt(x: Double): Double

/**
 * Computes the hyperbolic sine of the value [x].
 *
 * Special cases:
 *   - `sinh(NaN)` is `NaN`
 *   - `sinh(x)` is `+Inf` when `x > +Inf`
 *   - `sinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun sinh(x: Double): Double

/**
 * Computes the hyperbolic cosine of the value [x].
 *
 * Special cases:
 *   - `cosh(NaN)` is `NaN`
 *   - `cosh(+Inf)` is `+Inf`
 *   - `cosh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun cosh(x: Double): Double

/**
 * Computes the hyperbolic tangent of the value [x].
 *
 * Special cases:
 *   - `tanh(NaN)` is `NaN`
 *   - `tanh(+Inf)` is `+1.0`
 *   - `tanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
public expect fun tanh(x: Double): Double

/**
 * Computes the inverse hyperbolic sine of the value [x].
 *
 * Special cases:
 *   - `asinh(NaN)` is `NaN`
 *   - `asinh(x)` is `+Inf` when `x > +Inf`
 *   - `asinh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun asinh(x: Double): Double

/**
 * Computes the inverse hyperbolic cosine of the value [x].
 *
 * Special cases:
 *   - `acosh(NaN)` is `NaN`
 *   - `acosh(+Inf)` is `+Inf`
 *   - `acosh(-Inf)` is `-Inf`
 */
@SinceKotlin("1.2")
public expect fun acosh(x: Double): Double

/**
 * Computes the inverse hyperbolic tangent of the value [x].
 *
 * Special cases:
 *   - `atanh(NaN)` is `NaN`
 *   - `atanh(+Inf)` is `1.0`
 *   - `atanh(-Inf)` is `-1.0`
 */
@SinceKotlin("1.2")
public expect fun atanh(x: Double): Double

/**
 * Computes the arc cosine of the value [x]; the returned value is an angle in the range from `0.0` to `PI` radians.
 *
 * Special cases:
 *   - `acos(NaN)` is `NaN`, when `abs(x) > 1` or x is `NaN`
 *   - `acos(1.0)` is `0.0`
 */
@SinceKotlin("1.2")
public expect fun acos(x: Double): Double

/**
 * Computes the arc sine of the value [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 *
 * Special cases:
 *   - `asin(NaN)` is `NaN`, when `abs(x) > 1` or x is `NaN`
 *   - `asin(1.0)` is `0.0`
 */
@SinceKotlin("1.2")
public expect fun asin(x: Double): Double

/**
 * Computes the arc tangent of the value [y] by computing the arc tangent of the value [x] / [y]; the returned value is an angle in the range from `-PI` to `PI` radians.
 *
 * Special cases:
 *   - `atan(0.0)` is `0.0`
 *   - `atan(1.0)` is `+PI/2`
 *   - `atan(-1.0)` is `-PI/2`
 */
@SinceKotlin("1.2")
public expect fun atan(x: Double): Double

/**
 * Computes the arc tangent of the value [y] / [x]; the returned value is an angle in the range from `-PI/2` to `PI/2` radians.
 *
 * Special cases:
 *   - `atan2(0.0, 0.0)` is `0.0`
 *   - `atan2(0.0, x)` is `0.0` for `x > 0` and `PI` for `x < 0`
 *   - `atan2(-0.0, x)` is `-0.0` for `x > 0` and `-PI` for `x < 0`
 */
@SinceKotlin("1.2")
public expect fun atan2(y: Double, x: Double): Double
function for special cases.

```kotlin
@SinceKotlin("1.2")
public expect fun log10(x: Double): Double

/**
 * Computes the binary logarithm (base 2) of the value [x].
 * @see [ln] function for special cases.
 */
@SinceKotlin("1.2")
public expect fun log2(x: Double): Double

/**
 * Computes `ln(x + 1)`.
 * @see [ln] function
 */
@SinceKotlin("1.2")
public expect fun ln1p(x: Double): Double

/**
 * Rounds the given value [x] to an integer towards positive infinity.
 * @return the smallest double value that is greater than or equal to the given value [x] and is a mathematical integer.
 */
@SinceKotlin("1.2")
public expect fun ceil(x: Double): Double

/**
 * Rounds the given value [x] towards the closest integer with ties rounded towards even integer.
 */
@SinceKotlin("1.2")
public expect fun truncate(x: Double): Double

/**
 * Rounds the given value [x] towards zero.
 */
@SinceKotlin("1.2")
public expect fun floor(x: Double): Double

/**
 * Returns the absolute value of the given value [x].
 */
@SinceKotlin("1.2")
public expect fun abs(x: Double): Double

/**
 * Returns the sign of the given value [x]:
 *   - `-1.0` if the value is negative,
 *   - zero if the value is zero,
 *   - `1.0` if the value is positive
 */
@SinceKotlin("1.2")
public expect fun sign(x: Double): Double

/**
 * Returns the smaller of two values.
 */
@SinceKotlin("1.2")
public expect fun min(a: Double, b: Double): Double

/**
 * Returns the greater of two values.
 */
@SinceKotlin("1.2")
public expect fun max(a: Double, b: Double): Double
```

Double

```kotlin
/**
 * Raises this value to the power [x].
 * @see abs function
 */
@SinceKotlin("1.2")
public expect fun pow(x: Double): Double

/**
 * Raises this value to the integer power [n].
 * @see abs function
 */
@SinceKotlin("1.2")
public expect fun pow(n: Int): Double
```
\`x.nextTowards(y)` is `NaN` if either `x` or `y` are `NaN`\n\* \- \`x.nextTowards(x)` == \`x`\n
`x.roundToInt()` == `Int.MAX_VALUE` when `x > Int.MAX_VALUE`\n\* \- \`x.roundToInt()` == `Int.MIN_VALUE` when `x < Int.MIN_VALUE`\n
Throws `IllegalStateException` when this value is `NaN`\n\* `@SinceKotlin("1.2")`\n
`Double.roundToLong()` == `Long.MAX_VALUE` when `x >`\n\* `@SinceKotlin("1.2")`\n
// endregion

```
// region ================ Float Math

/** Computes the sine of the angle [x] given in radians.\n * Special cases:\n * \- `\sin(NaN)` is `NaN`\n * \- `\cos(NaN)` is `NaN`\n * `\sin(\pm\infty)` is `\pm1`\n * `\cos(\pm\infty)` is `0`\n */
@SinceKotlin("1.2")
public expect fun `sin(x: Float)`:
```

```
/** Computes the cosine of the angle [x] given in radians.\n * Special cases:\n * \- `\cos(NaN)` is `NaN`\n * \- `\sin(NaN)` is `NaN`\n * `\sin(\pm\infty)` is `\pm1`\n * `\cos(\pm\infty)` is `0`\n */
@SinceKotlin("1.2")
public expect fun `cos(x: Float)`:
```

```
/** Computes the tangent of the angle [x] given in radians.\n * Special cases:\n * `\tan(NaN)` is `NaN`\n * `\tan(\pm\infty)` is `\pm1`\n */
@SinceKotlin("1.2")
public expect fun `tan(x: Float)`:
```

```
/** Computes the hyperbolic sine of the value [x].\n * Special cases:\n * `\sinh(NaN)` is `NaN`\n * `\cosh(NaN)` is `NaN`\n * `\sinh(\pm\infty)` is `\pm\infty`\n */
@SinceKotlin("1.2")
public expect fun `sinh(x: Float)`:
```

```
/** Computes the hyperbolic cosine of the value [x].\n * Special cases:\n * `\cosh(NaN)` is `NaN`\n * `\sinh(NaN)` is `NaN`\n * `\cosh(\pm\infty)` is `\pm\infty`\n */
@SinceKotlin("1.2")
public expect fun `cosh(x: Float)`:
```

```
/** Computes the hyperbolic tangent of the value [x].\n * Special cases:\n * `\tanh(NaN)` is `NaN`\n * `\tanh(\pm\infty)` is \`\pm1`\n */
@SinceKotlin("1.2")
public expect fun `tanh(x: Float)`:
```

```
/** Computes the inverse hyperbolic sine of the value [x] when the returned value is positive.\n * Special cases:\n * `\asinh(NaN)` is `NaN`\n * `\asinh(\pm\infty)` is `\pm\infty`\n */
@SinceKotlin("1.2")
public expect fun `asinh(x: Float)`:
```

```
/** Computes the inverse hyperbolic cosine of the value [x] when the returned value is positive.\n * Special cases:\n * `\acosh(NaN)` is `NaN`\n * `\acosh(\pm\infty)` is `\pm\infty`\n */
@SinceKotlin("1.2")
public expect fun `acosh(x: Float)`:
```

```
/** Computes the inverse hyperbolic tangent of the value [x] when the returned value is positive.\n * Special cases:\n * `\atanh(NaN)` is `NaN`\n * `\atanh(\pm\infty)` is `\pm0.5`\n */
@SinceKotlin("1.2")
public expect fun `atanh(x: Float)`:
```

```
```
infinite
 * - returns `NaN` if any of arguments is `NaN` and the other is not infinite

@SinceKotlin("1.2") public expect fun hypot(x: Float, y: Float): Float Float\n\n
/**
 * Computes the positive square root of the value \([x, y]\). * 
 * - `sqrt(x) is NaN` when \(x < 0\) or \(x, y\) are `NaN`

@SinceKotlin("1.2") public expect fun sqrt(x: Float): Float

/**
 * Computes Euler's number \(e\) raised to the power of the value \(x\):

@SinceKotlin("1.2") public expect fun exp(x: Float): Float

/**
 * Computes `exp(x) - 1.0` * 
 * This function can be implemented to produce more precise result for \(x\) near zero.

@SinceKotlin("1.2") public expect fun expm1(x: Float): Float

/**
 * Computes the logarithm of the value \([x, y]\) to the given base. * 
 * Special cases:

@SinceKotlin("1.2") public expect fun log(x: Float, base: Float): Float

/**
 * Computes the natural logarithm (base \(E\)) of the value \(x\):

@SinceKotlin("1.2") public expect fun ln(x: Float): Float

/**
 * Computes the common logarithm (base 10) of the value \(x\):

@SinceKotlin("1.2") public expect fun log10(x: Float): Float

/**
 * Computes the binary logarithm (base 2) of the value \(x\):

@SinceKotlin("1.2") public expect fun log2(x: Float): Float

/**
 * Returns the absolute value of the given value \(x\):

@SinceKotlin("1.2") public expect fun abs(x: Float): Float

/**
 * Returns the sign of the given value \(x\):

@SinceKotlin("1.2") public expect fun sign(x: Float): Float

 /**
 * Returns the smaller of two values.

@SinceKotlin("1.2") public expect fun min(a: Float, b: Float): Float

 /**
 * Returns the greater of two values.

@SinceKotlin("1.2") public expect fun max(a: Float, b: Float): Float

 */

/**
 * Returns the sign of the given value \(x\):

@SinceKotlin("1.2") public expect fun sign(x: Float): Float

 */

/**
 * Raises this value to the power \(x\):

@SinceKotlin("1.2") public expect fun Float.pow(x: Float): Float
*\@SinceKotlin("1.2")\npublic expect fun Float.pow(n: Int): Float\n
/**
 * Returns the absolute value of this value.\n * \n * Special cases:\n * - `NaN.absoluteValue` is `NaN`\n * - `-1.0` if the value is negative,\n * - `0` if the value is zero,\n * - `1.0` if the value is positive.\n * \n * @see abs function\n */
*\@SinceKotlin("1.2")\npublic expect val Float.absoluteValue: Float\n
/**
 * Returns the sign of this value.\n * \n * Special cases:\n * - `NaN.sign` is `NaN`\n */
*\@SinceKotlin("1.2")\npublic expect val Float.sign: Float\n
/**
 * Rounds this [Float] value to the nearest integer and converts the result to [Int].\n * \n * Ties are rounded towards positive infinity.\n */
*\@SinceKotlin("1.2")\npublic expect fun Float.roundToInt(): Int\n
/**
 * Returns the absolute value of the given value [n].\n */
*\@SinceKotlin("1.2")\npublic expect fun abs(n: Int): Int\n
/**
 * Returns the smaller of two values.\n */
*\@SinceKotlin("1.2")\npublic expect fun min(a: Int, b: Int): Int\n
/**
 * Returns the greater of two values.\n */
*\@SinceKotlin("1.2")\npublic expect fun max(a: Int, b: Int): Int\n
/**
 * Returns the absolute value of this value.\n */
*\@SinceKotlin("1.2")\npublic expect val Int.absoluteValue: Int\n
/**
 * Returns the sign of this value:\n * \n * Special cases:\n * - `0` if the value is zero,\n */
*\@SinceKotlin("1.2")\npublic expect val Int.sign: Int\n
/**
 * Returns the absolute value of the given value [n].\n */
*\@SinceKotlin("1.2")\npublic expect fun abs(n: Long): Long\n
/**
 * Returns the smaller of two values.\n */
*\@SinceKotlin("1.2")\npublic expect fun min(a: Long, b: Long): Long\n
/**
 * Returns the greater of two values.\n */
*\@SinceKotlin("1.2")\npublic expect fun max(a: Long, b: Long): Long\n
endregion\n"
Open Source Used In IOx-FogDirector-AC18 1.17.0  2638
GA4D0C,EA5DxCB,Ca4DxCB,CAAX,GAA6C,UA5DvE,WA4DuE,6DAA5D,EA5DE,CA4DF,C/FjGjD,C;;Q+FiGA
,OACY,W;O;KA7BhB,CiFgAC,yB;MAAA,gB;MAAA,8B;MfHFA,wE;M+FgFA,qD;QAF7FW,kB/QfBd0,B,Q +FM9C,Q;QAAA,OAAK,0B;QAfF,OAAU;AAV,C;UAUU,mB;UCN,UAU,SB,BAAM,CAAN,CAV,Ck,BAAbB,
,sBAAY,GAAZ,C;UAkFID,U;UAjFne,W,F6JjJ+a=F7jgB,G/QfHbE, E+F5EgC,UjFJsB,UBaABu,B,CAAC,WAAY,
mBAAY,GAAZ,CaiFhD,kBAA6B,UBAEEj,C;WAIjFc,W;DAvC,eEjMfB,CiFmA,0F+C4EHlC,C;;+Q+F7EA,QA9E0,
W;O;KA6DxCB,CA4F0A0B,yB;MAAA,gB;MAAA,8B;MAAA,kE,Q;QAAA,OAAK,0B;QAfF,OAAU,CAAC
,C;UAUU,mB;UCN,UAU,SB,BAAM,CAAN,CAV,Ck,BAAbB,
,sBAAY,GAAZ,C;UAkFID,U;UAjFne,W,F6JjJ+a=F7jgB,G/QfHbE, E+F5EgC,UjFJsB,UBaABu,B,CAAC,WAAY,
1.237 e2fsprogs 1.44.1 1ubuntu1.3

1.237.1 Available under license:

This is the Debian GNU/Linux prepackaged version of the static EXT2 file system consistency checker (e2fsck.static). The EXT2 utilities were written by Theodore Ts'o <tytso@mit.edu> and Remy Card <card@masi.ibp.fr>.

Sources were obtained from http://sourceforge.net/projects/e2fsprogs

Packaging is Copyright (c) 2003-2006 Theodore Ts'o <tytso@mit.edu>
   Copyright (c) 1997-2003 Yann Dirson <dirson@debian.org>
   Copyright (c) 2001 Alcove <http://www.alcove.com/>
   Copyright (c) 1997 Klee Dienes
   Copyright (c) 1995-1996 Michael Nonweiler <mrn20@cam.ac.uk>

Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright notice:

This package, the EXT2 filesystem utilities, is protected by the GNU General Public License.


On Debian GNU systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2'.

This package was added to the e2fsprogs debian source package by Theodore Ts'o <tytso@mit.edu> on Sat Mar 15 15:33:37 EST 2003

It is part of the main e2fsprogs distribution, which can be found at:

http://sourceforge.net/projects/e2fsprogs

Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright:

Copyright (C) 1999, 2001 by Andries Brouwer
Copyright (C) 1999, 2000, 2003 by Theodore Ts'o
You are free to distribute this software under the terms of the GNU Lesser (Library) General Public License.

On Debian systems, the complete text of the GNU Lesser (Library) General Public License can be found in /usr/share/common-licenses/LGPL-2. This package was added to the e2fsprogs debian source package by Theodore Ts'o <tytso@mit.edu> on Fri Dec 14 22:24:35 EST 2007

It is part of the main e2fsprogs distribution, which can be found at:

http://sourceforge.net/projects/e2fsprogs

Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright:


Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright
notice, and the entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This is the Debian GNU/Linux prepackaged version of the translation files of the EXT2 file system utilities. The EXT2 utilities were written by Theodore Ts'o <tytso@mit.edu> and Remy Card <card@masi.ibp.fr>.

Sources were obtained from http://sourceforge.net/projects/e2fsprogs

Packaging is Copyright (c) 2003-2006 Theodore Ts'o <tytso@mit.edu>  
Copyright (c) 1997-2003 Yann Dirson <dirson@debian.org>  
Copyright (c) 2001 Alcove <http://www.alcove.com/>  
Copyright (c) 1997 Klee Dienes  
Copyright (c) 1995-1996 Michael Nonweiler <mrn20@cam.ac.uk>

Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright notice:

This package, the EXT2 filesystem utilities, is protected by the GNU General Public License.


On Debian GNU systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2`: */
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T. and the M.I.T. S.I.P.B. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. and the M.I.T. S.I.P.B. make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

*/
Index: tdbsa/tdb.c
===================================================================
--- tdbsa.orig/tdb.c
+++ tdbsa/tdb.c
@@ -4,11 +4,11 @@
Last Changed Date: 2007-06-06 20:14:06 -0400 (Wed, 06 Jun 2007)
*/
/
- Unix SMB/CIFS implementation.
+ trivial database library - standalone version

- trivial database library - private includes
-
- Copyright (C) Andrew Tridgell 2005
+ Copyright (C) Andrew Tridgell 1999-2005
+ Copyright (C) Jeremy Allison 2000-2006
+ Copyright (C) Paul 'Rusty' Russell 2000

** NOTE! The following LGPL license applies to the tdb
** library. This does NOT imply that all of Samba is released
This is the Debian GNU/Linux prepackaged version of the EXT2 file
system utilities (e2fsck, mke2fs, etc.). The EXT2 utilities were
written by Theodore Ts'o <tytso@mit.edu> and Remy Card <card@masi.ibp.fr>.

Sources were obtained from http://sourceforge.net/projects/e2fsprogs

Packaging is Copyright (c) 2003-2007 Theodore Ts'o <tytso@mit.edu>
   Copyright (c) 1997-2003 Yann Dirson <dirson@debian.org>
   Copyright (c) 2001 Alcove <http://www.alcove.com/>
   Copyright (c) 1997 Klee Dienes
   Copyright (c) 1995-1996 Michael Nonweiler <mrn20@cam.ac.uk>
Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright notice:

This package, the EXT2 filesystem utilities, are made available under the GNU General Public License version 2, with the exception of the lib/ext2fs and lib/e2p libraries, which are made available under the GNU Library General Public License Version 2, the lib/uuid library which is made available under a BSD-style license and the lib/et and lib/ss libraries which are made available under an MIT-style license.


On Debian GNU systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2'. The complete text of the GNU Library General Public License can be found in `/usr/share/common-licenses/LGPL-2'.

The license used for lib/et and lib/ss libraries is:

Copyright 1987 by the Student Information Processing Board of the Massachusetts Institute of Technology

Permission to use, copy, modify, and distribute this software and its documentation for any purpose is hereby granted, provided that the names of M.I.T. and the M.I.T. S.I.P.B. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. and the M.I.T. S.I.P.B. make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

The license used for lib/uuid is:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This package, the EXT2 filesystem utilities, are made available under the GNU Public License version 2, with the exception of the lib/ext2fs and lib/e2p libraries, which are made available under the GNU Library General Public License Version 2, the lib/uuid library which is made available under a BSD-style license and the lib/et and lib/ss libraries which are made available under an MIT-style license. Please see lib/uuid/COPYING for more details for the license for the files comprising the libuuid library, and the source file headers of the libet and libss libraries for more information.

The most recent officially distributed version can be found at http://e2fsprogs.sourceforge.net. If you need to make a distribution, that's the one you should use. If there is some reason why you'd like a more recent version that is still in ALPHA testing (i.e., either using the "WIP" test distributions or one from the hg or git repository from the development branch, please contact me (tytso@mit.edu) before you ship. The release schedules for this package are flexible, if you give me enough lead time.

Theodore Ts'o
23-June-2007

-----------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---------------------------------------------------------------------

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original
version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under
3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library.
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above
specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent license
would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Library General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
This package was added to the e2fsprogs debian source package by Theodore Ts'o <tytso@mit.edu> on Sat Mar 15 15:33:37 EST 2003
It is part of the main e2fsprogs distribution, which can be found at:

http://sourceforge.net/projects/e2fsprogs

Upstream Author: Theodore Ts'o <tytso@mit.edu>

Copyright:

Copyright (C) 1999, 2000, 2003, 2004 by Theodore Ts'o

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, and the entire permission notice in its entirety,
including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote
products derived from this software without specific prior
written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF
WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT
OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH
damage.

This is the Debian GNU/Linux prepackaged version of the ss
command-line interface parsing library. It is currently
distributed together with the EXT2 file system utilities, which are
otherwise packaged as "e2fsprogs".

This package was put together by Yann Dirson <dirson@debian.org>,
from sources obtained from a mirror of:
tsx-11.mit.edu/pub/linux/packages/ext2fs/

From the original distribution:

Copyright 1987, 1988 by the Student Information Processing Board
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T. and the M.I.T. S.I.P.B. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. and the M.I.T. S.I.P.B. make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

# This is a Makefile stub which handles the creation of BSD shared libraries.
# In order to use this stub, the following makefile variables must be defined.

# BSDLIB_VERSION = 1.0
# BSDLIB_IMAGE = libce
# BSDLIB_MYDIR = et
# BSDLIB_INSTALL_DIR = $(SHLIBDIR)

all:: image

real-subdirs:: Makefile
@echo "MKDIR pic"
@mkdir -p pic

BSD_LIB = $(BSDLIB_IMAGE).so.$(BSDLIB_VERSION)
BSDLIB_PIC_FLAG = -fpic

image::$(BSD_LIB)

$(BSD_LIB): $(OBJS)
(cd pic; ld -Bshareable -o $(BSD_LIB) $(LDFLAGS_SHLIB) $(OBJS))
$(MV) pic/$(BSD_LIB) .
$(RM) -f ../$(BSD_LIB)
(cd ..; $(LN) $(LINK_BUILD_FLAGS) `echo $(my_dir) | sed -e 's;lib/;;'`/$(BSD_LIB) $(BSD_LIB))

install-shlibs install:: $(BSD_LIB)
@echo "INSTALL_PROGRAM $(BSDLIB_INSTALL_DIR)/$(BSD_LIB)"
@$(INSTALL_PROGRAM) $(BSD_LIB) \
$(DESTDIR)$(BSDLIB_INSTALL_DIR)/$(BSD_LIB)
@-$(LDCONFIG)
install-strip: install
install-shlibs-strip: install-shlibs
uninstall-shlibs uninstall::
$(RM) -f $(DESTDIR)/$(BSDLIB_INSTALL_DIR)/$(BSD_LIB)
clean::
$(RM) -rf pic
$(RM) -f $(BSD_LIB)
$(RM) -f ../$(BSD_LIB)
This is the Debian GNU/Linux prepackaged version of the Common Error Description library. It is currently distributed together with the EXT2 file system utilities, which are otherwise packaged as "e2fsprogs".

This package was put together by Yann Dirson <dirson@debian.org>, from sources obtained from a mirror of:
tsx-11.mit.edu:/pub/linux/packages/ext2fs/

From the original distribution:

Copyright 1987, 1988 by the Student Information Processing Board of the Massachusetts Institute of Technology

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T. and the M.I.T. S.I.P.B. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. and the M.I.T. S.I.P.B. make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty. EXT2ED is hereby placed under the terms of the GNU General Public License. Follows the GNU license.

Gadi Oxman, August 1995

---------------------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any
The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free

Open Source Used In IOx-FogDirector-AC18 1.17.0 2729
Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
1.238 jetty-servlet 9.4.31.v20200723

1.238.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.

"Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.
Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
 exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided...
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and
* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2
The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.
Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.239 xmlsec-java 2.1.4
1.239.1 Available under license:
Apache Santuario - XML Security for Java
Copyright 1999-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

It was originally based on software copyright (c) 2001, Institute for

The development of this software was partly funded by the European Commission in the <WebSig> project in the ISIS Programme.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.240 encoder 1.2.2

1.240.1 Available under license:

Copyright (c) 2015 Jeff Ichnowski
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the OWASP nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.241 bash 4.4.18 2ubuntu1.2

1.241.1 Available under license:

GNU Free Documentation License
Version 1.3, 3 November 2008

<http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.
This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant
Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.
The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using
public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If
there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retile any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License
give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate
and independent documents or works, in or on a volume of a storage or
distribution medium, is called an "aggregate" if the copyright
resulting from the compilation is not used to limit the legal rights
of the compilation's users beyond what the individual works permit.
When the Document is included in an aggregate, this License does not
apply to the other works in the aggregate which are not themselves
derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these
copies of the Document, then if the Document is less than one half of
the entire aggregate, the Document's Cover Texts may be placed on
covers that bracket the Document within the aggregate, or the
electronic equivalent of covers if the Document is in electronic form.
Otherwise they must appear on printed covers that bracket the whole
aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may
distribute translations of the Document under the terms of section 4.
Replacing Invariant Sections with translations requires special
permission from their copyright holders, but you may include
translations of some or all Invariant Sections in addition to the
original versions of these Invariant Sections. You may include a
translation of this License, and all the license notices in the
Document, and any Warranty Disclaimers, provided that you also include
the original English version of this License and the original versions
of those notices and disclaimers. In case of a disagreement between
the translation and the original version of this License or a notice
or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements",
"Dedications", or "History", the requirement (section 4) to Preserve
its Title (section 1) will typically require changing the actual
title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense, or distribute it is void, and
will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license
from a particular copyright holder is reinstated (a) provisionally,
unless and until the copyright holder explicitly and finally
Terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.
"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.
GNU GENERAL PUBLIC LICENSE  
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA  
Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your  
freedom to share and change it. By contrast, the GNU General Public  
License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This  
General Public License applies to most of the Free Software  
Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by  
the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not  
price. Our General Public Licenses are designed to make sure that you  
have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it  
if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid  
anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you  
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether  
gratis or for a fee, you must give the recipients all the rights that  
you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and  
(2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain  
that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we  
want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further
restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for
infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is
available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law. If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own
removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of
that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the
terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or
author attributions in that material or in the Appropriate Legal
Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or
requiring that modified versions of such material be marked in
reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or
authors of the material; or

e) Declining to grant rights under trademark law for use of some
trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims
Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is

owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.
in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that oblige you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered
version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
@c The GNU Free Documentation License.
@center Version 1.3, 3 November 2008
@c This file is intended to be included within another document,
c hence no sectioning command or @node.
@display
@uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
@end display
@enumerate 0
@item
PREAMBLE

The purpose of this License is to make a manual, textbook, or other
functional and useful document @dfn{free} in the sense of freedom: to
assure everyone the effective freedom to copy and redistribute it,
with or without modifying it, either commercially or noncommercially.
Secondarily, this License preserves for the author and publisher a way
to get credit for their work, while not being considered responsible
for modifications made by others.

This License is a kind of ``copyleft'', which means that derivative
works of the document must themselves be free in the same sense. It
complements the GNU General Public License, which is a copyleft
license designed for free software.

We have designed this License in order to use it for manuals for free
software, because free software needs free documentation: a free
program should come with manuals providing the same freedoms that the
software does. But this License is not limited to software manuals;
it can be used for any textual work, regardless of subject matter or
whether it is published as a printed book. We recommend this License
principally for works whose purpose is instruction or reference.
@item
APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that
contains a notice placed by the copyright holder saying it can be
distributed under the terms of this License. Such a notice grants a
world-wide, royalty-free license, unlimited in duration, to use that
work under the conditions stated herein. The "Document", below,
refers to any such manual or work. Any member of the public is a
licensee, and is addressed as "you". You accept the license if you
copy, modify or distribute the work in a way requiring permission
under copyright law.

A "Modified Version" of the Document means any work containing the
Document or a portion of it, either copied verbatim, or with
modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section
of the Document that deals exclusively with the relationship of the
publishers or authors of the Document to the Document's overall
subject (or to related matters) and contains nothing that could fall
directly within that overall subject. (Thus, if the Document is in
part a textbook of mathematics, a Secondary Section may not explain
any mathematics.) The relationship could be a matter of historical
connection with the subject or with related matters, or of legal,
commercial, philosophical, ethical or political position regarding
them.

The "Invariant Sections" are certain Secondary Sections whose titles
are designated, as being those of Invariant Sections, in the notice
that says that the Document is released under this License. If a
section does not fit the above definition of Secondary then it is not
allowed to be designated as Invariant. The Document may contain zero
Invariant Sections. If the Document does not identify any Invariant
Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed,
as Front-Cover Texts or Back-Cover Texts, in the notice that says that
the Document is released under this License. A Front-Cover Text may
be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy,
represented in a format whose specification is available to the
general public, that is suitable for revising the document
straightforwardly with generic text editors or (for images composed of
pixels) generic paint programs or (for drawings) some widely available
drawing editor, and that is suitable for input to text formatters or
for automatic translation to a variety of formats suitable for input
to text formatters. A copy made in an otherwise Transparent file
format whose markup, or absence of markup, has been arranged to thwart
or discourage subsequent modification by readers is not Transparent.
An image format is not Transparent if used for any substantial amount
of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain 
@sc{ascii} without markup, Texinfo input format, La@TeX{} input 
format, @acronym{SGML} or @acronym{XML} using a publicly available 
@acronym{DTD}, and standard-conforming simple @acronym{HTML}, 
PostScript or @acronym{PDF} designed for human modification. Examples 
of transparent image formats include @acronym{PNG}, @acronym{XCF} and 
@acronym{JPG}. Opaque formats include proprietary formats that can be 
read and edited only by proprietary word processors, @acronym{SGML} or 
@acronym{XML} for which the @acronym{DTD} and/or processing tools are 
not generally available, and the machine-generated @acronym{HTML}, 
PostScript or @acronym{PDF} produced by some word processors for 
output purposes only.

The "Title Page" means, for a printed book, the title page itself, 
plus such following pages as are needed to hold, legibly, the material 
this License requires to appear in the title page. For works in 
formats which do not have any title page as such, "Title Page" means 
the text near the most prominent appearance of the work's title, 
preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies 
of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose 
title either is precisely XYZ or contains XYZ in parentheses following 
text that translates XYZ in another language. (Here XYZ stands for a 
specific section name mentioned below, such as "Acknowledgements", 
"Dedications", "Endorsements", or "History"). To "Preserve the Title" 
of such a section when you modify the Document means that it remains a 
section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which 
states that this License applies to the Document. These Warranty 
Disclaimers are considered to be included by reference in this 
License, but only as regards disclaiming warranties; any other 
implication that these Warranty Disclaimers may have is void and has 
no effect on the meaning of this License.

@item
VERBATIM COPYING

You may copy and distribute the Document in any medium, either 
commercially or noncommercially, provided that this License, the 
copyright notices, and the license notice saying this License applies 
to the Document are reproduced in all copies, and that you add no other 
conditions whatsoever to those of this License. You may not use
technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

@item
COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

@item
MODIFICATIONS
You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

@enumerate
@item
Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

@item
List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

@item
State on the Title page the name of the publisher of the Modified Version, as the publisher.

@item
Preserve all the copyright notices of the Document.

@item
Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

@item
Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

@item
Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled `"History"`, Preserve its Title, and add
to it an item stating at least the title, year, new authors, and
publisher of the Modified Version as given on the Title Page. If
there is no section Entitled ``History'' in the Document, create one
stating the title, year, authors, and publisher of the Document as
given on its Title Page, then add an item describing the Modified
Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for
public access to a Transparent copy of the Document, and likewise
the network locations given in the Document for previous versions
it was based on. These may be placed in the ``History'' section.
You may omit a network location for a work that was published at
least four years before the Document itself, or if the original
publisher of the version it refers to gives permission.

@item
For any section Entitled ``Acknowledgements'' or ``Dedications'', Preserve
the Title of the section, and preserve in the section all the
substance and tone of each of the contributor acknowledgements and/or
dedications given therein.

@item
Preserve all the Invariant Sections of the Document,
unaltered in their text and in their titles. Section numbers
or the equivalent are not considered part of the section titles.

@item
Delete any section Entitled ``Endorsements''. Such a section
may not be included in the Modified Version.

@item
Do not retile any existing section to be Entitled ``Endorsements'' or
to conflict in title with any Invariant Section.

@item
Preserve any Warranty Disclaimers.
@end enumerate

If the Modified Version includes new front-matter sections or
appendices that qualify as Secondary Sections and contain no material
copied from the Document, you may at your option designate some or all
of these sections as invariant. To do this, add their titles to the
list of Invariant Sections in the Modified Version's license notice.
These titles must be distinct from any other section titles.

You may add a section Entitled ``Endorsements'', provided it contains
nothing but endorsements of your Modified Version by various
parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

@item
COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements.”

@item
COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this
License in the various documents with a single copy that is included in
the collection, provided that you follow the rules of this License for
verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute
it individually under this License, provided you insert a copy of this
License into the extracted document, and follow this License in all
other respects regarding verbatim copying of that document.

@item
AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate
and independent documents or works, in or on a volume of a storage or
distribution medium, is called an "aggregate" if the copyright
resulting from the compilation is not used to limit the legal rights
of the compilation's users beyond what the individual works permit.
When the Document is included in an aggregate, this License does not
apply to the other works in the aggregate which are not themselves
derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these
copies of the Document, then if the Document is less than one half of
the entire aggregate, the Document's Cover Texts may be placed on
covers that bracket the Document within the aggregate, or the
electronic equivalent of covers if the Document is in electronic form.
Otherwise they must appear on printed covers that bracket the whole
aggregate.

@item
TRANSLATION

Translation is considered a kind of modification, so you may
distribute translations of the Document under the terms of section 4.
Replacing Invariant Sections with translations requires special
permission from their copyright holders, but you may include
translations of some or all Invariant Sections in addition to the
original versions of these Invariant Sections. You may include a
translation of this License, and all the license notices in the
Document, and any Warranty Disclaimers, provided that you also include
the original English version of this License and the original versions
of those notices and disclaimers. In case of a disagreement between
the translation and the original version of this License or a notice
or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements",
"Dedications", or "History", the requirement (section 4) to Preserve
its Title (section 1) will typically require changing the actual
TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this
Copyright (C) @var{year}  @var{your name}.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts. A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
@end group
@end smallexample

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts,
replace the ```with@dots{}Texts.'' line with this:

@smallexample
@group
with the Invariant Sections being @var{list their titles}, with
the Front-Cover Texts being @var{list}, and with the Back-Cover Texts
being @var{list}.
@end group
@end smallexample

If you have Invariant Sections without Cover Texts, or some other
combination of the three, merge those two alternatives to suit the
situation.

If your document contains nontrivial examples of program code, we
recommend releasing these examples in parallel under your choice of
free software license, such as the GNU General Public License,
to permit their use in free software.

@C Local Variables:
@c ispell-local-pdict: "ispell-dict"
@c End:
Unless otherwise stated, all files in this directory are Copyright (C)
Free Software Foundation, Inc.

See the file COPYING in the bash distribution root directory for copying
and usage restrictions.

The file ifs-posix.tests is Copyright (C) 2005 Glen Fowler.

1.242 ehcache 2.4.0
1.242.1 Available under license :
   Apache License
   Version 2.0, January 2004
   http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to
You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section)
patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such
license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was
submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit)
alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent
infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date
such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any
medium, with or without modifications, and in Source or Object form, provided that You meet the following
conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark,
and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of
the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute
must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text
file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the
Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices
normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an
addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed
as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license
terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works
as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions
stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product
names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work
(and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or
conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR
PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and
assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or
otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing,
shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or
consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of
such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may
choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or
rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf
and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend,
and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by
reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by
brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and
description of purpose be included on the same "printed page" as the copyright notice for easier identification within
third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
1.243 jc-tools-core 3.0.0

1.243.1 Available under license:
No license file was found, but licenses were detected in source scan.

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <artifactId>jc-tools-core</artifactId>
  <groupId>org.jctools</groupId>
  <version>3.0.0</version>
  <name>Java Concurrency Tools Core Library</name>
  <description>Java Concurrency Tools Core Library</description>
  <packaging>bundle</packaging>

  <dependencies>
    <dependency>
      <groupId>org.hamcrest</groupId>
      <artifactId>hamcrest-all</artifactId>
      <version>${hamcrest.version}</version>
      <scope>test</scope>
    </dependency>

    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>

    <dependency>
      <groupId>com.google.guava</groupId>
      <artifactId>guava-testlib</artifactId>
    </dependency>
  </dependencies>
</project>
```
<version>${guava-testlib.version}</version>
<scope>test</scope>
</dependency>
</dependencies>

<build>
<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-surefire-plugin</artifactId>
<version>3.0.0-M3</version>
<configuration>
<includes>
<include>*</include>
</includes>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.felix</groupId>
<artifactId>maven-bundle-plugin</artifactId>
<version>4.2.1</version>
<extensions>true</extensions>
<configuration>
<instructions>
<Import-Package>sun.misc;resolution:=optional</Import-Package>
</instructions>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
<version>3.2.0</version>
<executions>
<execution>
<id>attach-sources</id>
<phase>verify</phase>
<goals>
<goal>jar-no-fork</goal>
</goals>
</execution>
</executions>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
<version>3.1.1</version>
<configuration>
<additionalOptions>
-Xdoclint:none</additionalOption>
</configuration>
</plugin>
<additionalOptions/>
<source>8</source>
</configuration>
<executions>
<execution>
<id>attach-javadocs</id>
<goals>
<goal>jar</goal>
</goals>
</execution>
</executions>
</plugin>
</plugins>
</build>

<distributionManagement>
<repository>
<id>bintray-jctools-jctools</id>
<name>jctools-jctools</name>
<url>https://api.bintray.com/maven/jctools/jctools/jctools-core/;publish=1</url>
</repository>
</distributionManagement>

<url>https://github.com/JCTools</url>
<inceptionYear>2013</inceptionYear>

<licenses>
<license>
<name>Apache License, Version 2.0</name>
<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
<distribution>repo</distribution>
</license>
</licenses>

<scm>
<url>https://github.com/JCTools/JCTools</url>
<connection>scm:git:https://github.com/JCTools/JCTools</connection>
<tag>HEAD</tag>
</scm>

<developers>
<developer>
<url>https://github.com/nitsanw</url>
</developer>
<developer>
<url>https://github.com/mjpt777</url>
</developer>
</developers>
<url>https://github.com/RichardWarburton</url>
</developer>
<developer>
<url>https://github.com/kay</url>
</developer>
<developer>
<url>https://github.com/franz1981</url>
</developer>
</developers>

<prerequisites>
<maven>3.5.0</maven>
</prerequisites>

<properties>
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
<java.version>1.6</java.version>
<java.test.version>1.8</java.test.version>

<maven.compiler.source>${java.version}</maven.compiler.source>
<maven.compiler.target>${java.version}</maven.compiler.target>
<maven.compiler.testSource>${java.test.version}</maven.compiler.testSource>
<maven.compiler.testTarget>${java.test.version}</maven.compiler.testTarget>

<hamcrest.version>1.3</hamcrest.version>
<junit.version>4.12</junit.version>
<guava-testlib.version>21.0</guava-testlib.version>
</properties>
</project>

Found in path(s):
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3-0-0-sources-jar/META-INF/maven/org.jctools/jctools-core/pom.xml
No license file was found, but licenses were detected in source scan.

/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/MessagePassingQueueUtil.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/util/UnsafeJvmInfo.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/MpscAtomicArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/MpscLinkedAtomicQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/MpmcAtomicArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/SpscGrowableArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/SpscArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/maps/ConcurrentAutoTable.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/maps/NonBlockingIdentityHashMap.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/MpscGrowableArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/util/Pow2.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/spec/ConcurrentQueueSpec.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/BaseLinkedQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/util/InternalAPI.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/MpmcUnboundedXaddArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/SpscLinkedQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/SpscChunkedAtomicArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/spec/Ordering.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/spec/Preference.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/maps/AbstractEntry.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/MpscGrowableAtomicArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/atomic/BaseMpscLinkedAtomicArrayQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/MpscLinkedQueue.java
* /opt/cola/permits/1257637630_1642977964.98/0/jctools-core-3.0.0-sources.jar/org/jctools/queues/package-info.java
1.244 libwindheimdal 7.5.0+dfsg-1

1.244.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hsgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Please see info documentation for the complete list of licenses.
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
lib/wind/rfc3490.txt
lib/wind/rfc3491.txt
lib/wind/rfc4013.txt
lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in /usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone
# and Telegraph Company or of the Regents of the University of California.
#.
# Permission is granted to anyone to use this software for any purpose on
# any computer system, and to alter it and redistribute it freely, subject
# to the following restrictions:
# .
# 1. The authors are not responsible for the consequences of use of this
# software, no matter how awful, even if they arise from flaws in it.
# .
# 2. The origin of this software must not be misrepresented, either by
# explicit claim or by omission. Since few users ever read sources,
# credits must appear in the documentation.
# .
# 3. Altered versions must be plainly marked as such, and must not be
# misrepresented as being the original software. Since few users
# ever read sources, credits must appear in the documentation.
# .
# 4. This notice may not be removed or altered.

@macro copynext{}  
@vskip 20pt plus 1fil  
@end macro

@macro copyrightstart{}  
@end macro

@macro copyrightend{}  
@end macro

@node Copyrights and Licenses, , Acknowledgments, Top  
@comment node-name, next, previous, up  
@appendix Copyrights and Licenses

@heading Kungliga Tekniska Hgskolan

@copyrightstart  
@verbatim

Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

@end verbatim  
@copyrightend
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Massachusetts Institute of Technology

The parts of the libtelnet that handle Kerberos.

@verbatim

Copyright (C) 1990 by the Massachusetts Institute of Technology

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

@end verbatim
The parts of the libroken, most of libtelnet, telnet, ftp, and popper.

Copyright (c) 1988, 1990, 1993
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

libedit
Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim

@end verbatim
@copynext

@heading Doug Rabson
GSS-API mechglue layer.

@verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
@end verbatim
@copynext

@heading PADL Software Pty Ltd

@table @asis
@item GSS-API CFX, SPNEGO, naming extensions, API extensions.
@item KCM credential cache.
@item HDB LDAP backend.
@end table

@verbatim
Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@verbatim
Copyright (c) 2006,2007
NTT (Nippon Telegraph and Telephone Corporation) . All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.
vis.c in libroken

@verbatim

Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS
``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
@end verbatim
@copynext
@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim

rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>
This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim
Copyright (c) 2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require
a specific license from the United States Government. It is the
responsibility of any person or organization contemplating export to
obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and
distribute this software and its documentation for any purpose and
without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that copyright notice and
this permission notice appear in supporting documentation, and that
the name of Apple Inc. not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior
permission. Apple Inc. makes no representations about the suitability of
this software for any purpose. It is provided "as is" without express
or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim
@copynext
@heading Richard Outerbridge

DES core in libhcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec’s THINK (Lightspeed) C by Richard Outerbridge.
Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau,
for humouring me on.

(GEnie : OUTER; CIS : [71755,204]) Graven Imagery, 1992.

@end verbatim

@copynext

@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

- Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext
@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim
Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
1.245 jmte 4.0.0
1.245.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and limitations under the License.

1.246 sensible-utils 0.0.12

1.246.1 Available under license :

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: sensible-utils
Upstream-Contact: Anibal Monsalve Salazar <anibal@debian.org>
Source: https://alioth.debian.org/projects/collab-maint/sensible-utils.git

Files: *
Copyright: 2002-2009, Clint Adams <schizo@debian.org>
2010- Anibal Monsalve Salazar <anibal@debian.org>
2012, David Prvot <taffit@debian.org>
2013, Thorsten Glaser
2017, Jrmy Bobbio
2017, Ximin Luo
2017- Bastien Roucaris <rouca@debian.org>
License: GPL-2+

Files: sensible-editor*
Copyright: 1997, Guy Maor
2010- Anibal Monsalve Salazar <anibal@debian.org>
License: GPL-2+
Files: sensible-pager*
Copyright: 1997, 1998, Guy Maor
2004, Clint Adams
2010- Anibal Monsalve Salazar <anibal@debian.org>
License: GPL-2+

Files: sensible-browser*
Copyright: 2002, Joey Hess
2010- Anibal Monsalve Salazar <anibal@debian.org>
License: GPL-2+

Files: select-editor*
Copyright: 2009, Dustin Kirkland <kirkland@canonical.com>
2010- Anibal Monsalve Salazar <anibal@debian.org>
License: GPL-2+

Files: man/Makefile.am
man/utf8toman.sed
Copyright: 2012-2017, Guillaume Jover
License: GPL-2+
Comment: Part of this are copied from dpkg

Files: man/po4a/cs*
Copyright: 2012, Michal Simunek
License: GPL-2+

Files: man/po4a/de*
Copyright: 2011, Helge Kreutzmann <debian@helgefjell.de>
License: GPL-2+

Files: man/po4a/fr*
Copyright: Nicolas Francois <nicolas.francois@centralliens.net>
License: GPL-2+

Files: man/po4a/es*
Copyright: 2010-2012, Omar Campagne
License: GPL-2+

Files: man/po4a/it*
Copyright: 2012, Beatrice Torracca
License: GPL-2+

Files: man/po4a/ja*
Copyright: 2010, Kurasawa Nozomu
License: GPL-2+
Files: man/po4a/pl*
Copyright: 2004, 2010, Robert Luberda <robert@debian.org>.
License: GPL-2+

Files: man/po4a/pt*
Copyright: 2014, Amrico Monteiro <a_monteiro@gmx.com>
License: GPL-2+

Files: aclocal.m4
License: All-permissive

Files: *Makefile.in
License: All-permissive

Files: configure
License: configure
This configure script is free software; the Free Software Foundation

gives unlimited permission to copy, distribute and modify it.

Files: build-aux/missing
Copyright: 1996-2014, Free Software Foundation, Inc.
License: GPL-2+

Files: build-aux/install-sh
Copyright: 1994 X Consortium
License: installsh
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNE-
CTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not
be used in advertising or otherwise to promote the sale, use or other deal-
Copyright (c) 1996, 2013 Oracle and/or its affiliates. All rights reserved. See the file LICENSE for redistribution information.
This software is copyrighted by Christian Werner <chw@ch-werner.de> and other authors. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement,
license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

This software is copyrighted by Christian Werner <chw@ch-werner.de> and others. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

NOTE

The license is based on the zlib/libpng license. For more details see http://www.opensource.org/licenses/zlib-license.html. The intent of the
license is to:

- keep the license as simple as possible
- encourage the use of CuTest in both free and commercial applications and libraries
- keep the source code together
- give credit to the CuTest contributors for their work

If you ship CuTest in source form with your source distribution, the following license document must be included with it in unaltered form.
If you find CuTest useful we would like to hear about it.

LICENSE

Copyright (c) 2003 Asim Jalil

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

This package was debianized by Sam Clegg <samo@debian.org> on Tue, 25 Jul 2006 11:43:45 +0100.

It was downloaded from <http://www.ch-werner.de/sqliteodbc/>

Upstream Author: Christian Werner <chw@ch-werner.de>

Copyright: Copyright (c) 2001-2011 Christian Werner <chw@ch-werner.de>
             OS/2 Port Copyright (c) 2004 Lorne R. Sunley <lsunley@mb.sympatico.ca>

License:

This software is copyrighted by Christian Werner <chw@ch-werner.de> and other authors. The following terms apply to all files associated
with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

The Debian packaging is (C) 2006, Sam Clegg <samo@debian.org> and is licensed under the GPL-3, see `/usr/share/common-licenses/GPL-3'.

/*
* $Id$
*/

The following is the license that applies to this copy of the Berkeley DB software. For a license to use the Berkeley DB software under conditions other than those described here, or to purchase support for this software, please contact Oracle at berkeleydb-info_us@oracle.com.

/*
* Copyright (c) 1990, 2013 Oracle and/or its affiliates. All rights reserved.
* 
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
*/

Open Source Used In IOx-FogDirector-AC18 1.17.0 2824
* 3. Redistributions in any form must be accompanied by information on
* how to obtain complete source code for the DB software and any
* accompanying software that uses the DB software. The source code
* must either be included in the distribution or be available for no
* more than the cost of distribution plus a nominal fee, and must be
* freely redistributable under reasonable conditions. For an
* executable file, complete source code means the source code for all
* modules it contains. It does not include source code for modules or
* files that typically accompany the major components of the operating
* system on which the executable file runs.
*
* THIS SOFTWARE IS PROVIDED BY ORACLE `AS IS'' AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
* WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR
* NON-INFRINGEMENT, ARE DISCLAIMED. IN NO EVENT SHALL ORACLE BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
* BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
* WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
* OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
* IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
/*
* Copyright (c) 1990, 1993, 1994, 1995
* The Regents of the University of California. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the University nor the names of its contributors
*    may be used to endorse or promote products derived from this software
*    without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS `AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*/
/*
* Copyright (c) 1995, 1996
* The President and Fellows of Harvard University. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the University nor the names of its contributors
* may be used to endorse or promote products derived from this software
* without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY HARVARD AND ITS CONTRIBUTORS ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL HARVARD OR ITS CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*/
=-=-=-=-=-=-=-=-=-=-=-
/***
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

**1.248 txw2 2.3.1**

**1.248.1 Available under license:**

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2005-2018 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You * may not use...
this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License*. Found license 'General Public License 2.0' in *'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License // This class is in the Public Domain. NO WARRANTY! * Public Domain, and comes with <strong>NO WARRANTY</strong>.</em>*

Found license 'General Public License 2.0' in *'Copyright (c) 2005-2017 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License // NO WARRANTY! This class is in the public domain.'

1.249 go-runtime 1.15.3
1.249.1 Available under license :

# This is the official list of people who can contribute
# (and typically have contributed) code to the Go repository.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# Names should be added to this file only after verifying that
# the individual or the individual's organization has agreed to
# the appropriate Contributor License Agreement, found here:
#
# http://code.google.com/legal/individual-cla-v1.0.html
#
# The agreement for individuals can be filled out on the web.
#
# Names should be added to this file like so:
# Individual's name <submission email address>
# Individual's name <submission email address> <email2> <emailN>
#
# An entry with multiple email addresses specifies that the
# first address should be used in the submit logs and
# that the other addresses should be recognized as the
# same person when interacting with Gerrit.
#
# Please keep the list sorted.

Aamir Khan <syst3m.w0rm@gmail.com>
Aaron Beitch <aaronb@arista.com>
Alexander Menzhinsky <amenzhinsky@gmail.com>
Alexander Morozov <lk4d4math@gmail.com>
Alexander Neumann <alexander@bumpern.de>
Alexander Nohe <alex.nohe427@gmail.com>
Alexander Orlov <alexander.orlov@loxa1.net>
Alexander Pantyukhin <apantykhin@gmail.com>
Alexander Polcyn <apolcyn@google.com>
Alexander Rakoczy <alex@golang.org>
Alexander Reece <awreece@gmail.com>
Alexander Surma <surma@surmair.de>
Alexander Zhavnerchik <alex.vizor@gmail.com>
Alexander Zillion <alex@alexzillion.com>
Alexander Zolotov <goldifit@gmail.com>
Alexandr Mayorskiy <a.mayorskiy@corp.mail.ru>
Alexandre Cesaro <alexandre.cesaro@gmail.com>
Alexandre Fiori <fiorix@gmail.com>
Alexandre Maari <draeron@gmail.com>
Alexandre Normand <alexandre.normand@gmail.com>
Alexandre Parentea <aubonbeurre@gmail.com>
Alexandre Viau <alexandre@alexandreviau.net>
Alexandru Mooti <brtsnr@gmail.com>
Alexei Sholik <alcosholik@gmail.com>
Alexey Alexandrov <aalexand@gmail.com>
Alexey Borzenkov <snaury@gmail.com>
Alexey Naidonov <aleexy.naidyonov@gmail.com>
Alexey Neganov <neganovalexey@gmail.com>
Alexey Palazhchenko <aleexy.palazhchenko@gmail.com>
Alexey Semenyuk <alexsemenyuk88@gmail.com>
Alexis Hildebrandt <surryhill@gmail.com>
Alexis Hunt <lexer@google.com>
Alexis Imperial-Legrand <ail@google.com>
Ali Farooq <alifarooq0@pm.me>
Ali Rizvi-Santiago <arizvisa@gmail.com>
Aliaksandr Valialkin <valyala@gmail.com>
Alif Rachmawadi <subosito@gmail.com>
Allan Simon <allan.simon@supinfo.com>
Allen Li <ayatane@gmail.com>
Alok Menghrajani <alok.menghrajani@gmail.com>
Aman Gupta <aman@tm1.net>
Amarjeet Anand <amarjeetandSingh@gmail.com>
Amir Mohammad Saeed <amir@gluegadget.com>
Amr Mohammed <merodiro@gmail.com>
Amrut Joshi <amrut.joshi@gmail.com>
An Long <aikan1988@gmail.com>
An Xiao <hac@zju.edu.cn>
Anand K. Mistry <anand@mistry.ninja>
Anders Pearson <anders@columbia.edu>
Anderson Queiroz <contato@Andersonq.eti.br>
Ariel Mashraki <ariel@mashraki.co.il>
Arkadi Pyuro <arkadi@google.com>
Arlo Breault <arloolra@gmail.com>
Arnaud Ysmal <arnaud.ysmal@gmail.com>
Arne Hormann <arnehormann@gmail.com>
Arnout Engelen <arnout@bztt.net>
Aron Nopanen <aron.nopanen@gmail.com>
Artem Alekseev <artem.alekseev@intel.com>
Artem Kolin <artemkaxboy@gmail.com>
Arthur Fabre <arthur@arthurfabre.com>
Arthur Khashaev <arthur@khashaev.ru>
Artyom Pervukhin <artyom.pervukhin@gmail.com>
Arvindh Rajesh Tamilmani <art@a-30.net>
Ashish Gandhi <ag@ashishgandhi.org>
Asim Shankar <asimshankar@gmail.com>
Atin Malaviya <amalaviy@akamai.com>
Ato Araki <ato.araki@gmail.com>
Audrey Lim <audreylh@gmail.com>
Audrius Butkevicius <audrius.butkevicius@gmail.com>
Augusto Roman <aroman@gmail.com>
Aulus Egnatius Varialus <varialus@gmail.com>
Aurlien Rainone <aurelien.rainone@gmail.com>
Aurlio A. Heckert <aurium@gmail.com>
Austin Clements <austin@google.com> <aclements@csail.mit.edu>
Avi Flax <avi@timehop.com>
awaw fumin <awawfumin@gmail.com>
Awn Umar <awn@cryptolosophy.io>
Axel Wagner <axel.wagner.hh@googlemail.com>
Ayan George <ayan@ayan.net>
Ayanamist Yang <ayanamist@gmail.com>
Ayke van Laethem <aykevanlaethem@gmail.com>
Aymerrick Jhanne <aymerick@jehanne.org>
Azat Kaumov <kaumov.a.r@gmail.com>
Baiju Muthukadan <baiju.m.mail@gmail.com>
Balaram Makam <bmakam.qdt@qualcommdatcenter.com>
Balazs Lecz <leczb@google.com>
Baokun Lee <nototon@gmail.com>
Barnaby Keene <accounts@southcla.ws>
Bartosz Grzybowski <melkorm@gmail.com>
Bartosz Oler <brtsz@google.com>
Bastian Ike <bastian.ike@gmail.com>
Ben Burkert <ben@benburkert.com>
Ben Eitzen <eitzenb@golang.org>
Ben Fried <ben.fried@gmail.com>
Ben Haines <bhainesva@gmail.com>
Ben Hoyt <benhoyt@gmail.com>
Ben Laurie <ben@links.org> <benl@google.com>
Ben Lubar <ben.lubar@gmail.com>
Open Source Used In IOX-FogDirector-AC18 1.17.0  2839

Cyrill Schumacher <cyrill@schumacher.fm>
Daisuke Fujita <dtanshi45@gmail.com>
Daisuke Suzuki <daisuzu@gmail.com>
Daker Fernandes Pinheiro <daker.fernandes.pinheiro@intel.com>
Damian Gryski <dgryski@gmail.com>
Damien Lespiau <damien.lespiau@gmail.com>  <damien.lespiau@intel.com>
Damien Mathieu <42@dmathieu.com>
Damien Neil <dneil@google.com>
Damien Tournoud <damien@platform.sh>
Dan Ballard <dan@mindstab.net>
Dan Caddigan <goldcaddy77@gmail.com>
Dan Callahan <dan.callahan@gmail.com>
Dan Harrington <harringtond@google.com>
Dan Jacques <dnj@google.com>
Dan Johnson <computerdruid@gmail.com>
Dan Peterson <dpiddy@gmail.com>
Dan Pupius <dan@medium.com>
Dan Scales <danscales@gmail.com>
Dan Sinclair <dan.sinclair@gmail.com>
Daniel Cormier <danielc@knowbe4.com>
Danil de Kok <me@danieldk.eu>
Daniel Fleischman <danielfleischman@gmail.com>
Daniel Ingram <ingramds@appstate.edu>
Daniel Johansson <dajo2002@gmail.com>
Daniel Kerwin <d.kerwin@gni.net>
Daniel Krech <eikeon@eikeon.com>
Daniel Kurom <rdkumor@gmail.com>
Daniel Langner <s8572327@gmail.com>
Daniel Lidn <daniel.liden.87@gmail.com>
Daniel Lublin <daniel@lublin.se>
Daniel Mangum <georgedanielmangum@gmail.com>
Daniel Mart <mvdan@mvdan.cc>
Daniel Morsing <daniel.morsing@gmail.com>
Daniel Nadasi <dnadasi@gmail.com>
Daniel Nephin <dnephin@gmail.com>
Daniel Ortiz Pereira da Silva <daniel.particular@gmail.com>
Daniel Skinner <daniel@dasa.cc>
Daniel Speichert <daniel@speichert.pl>
Daniel Theophanes <kardianos@gmail.com>
Daniel Upton <daniel@floppy.co>
Daniela Petruzalek <daniela.petruzalek@gmail.com>
Danish Dua <danishdua@gmail.com>
Danish Prakash <grafitykoncept@gmail.com>
Danny Rosseau <daniel.roesseau@gmail.com>
Daria Kolistatova <daria.kolistatova@intel.com>
Darien Raymond <admin@v2ray.com>
Darren Elwood <dareer@textnode.com>
Darren Grant <dareer.e.grant@gmail.com>
Darren McCleary <darren.rmc@gmail.com>
Darshan Parajuli <parajulidarshan@gmail.com>
Datong Sun <dndx@idndx.com>
Dave Borowitz <dborowitz@google.com>
Dave Bort <dbort@golang.org>
Dave Cheney <dave@cheney.net>
Dave Day <djid@golang.org>
Dave Grijalva <dgrijalva@ngmoco.com>
Dave MacFarlane <driusan@gmail.com>
Dave Russell <forfunscake@gmail.com>
David Anderson <danderson@google.com>
David Barnett <dbarnett@google.com>
David Benjamin <davidben@google.com>
David Bond <davidsbond93@gmail.com>
David Brophy <dave@brophy.uk>
David Brgin <676c7473@gmail.com>
David Calavera <david.calavera@gmail.com>
David Carlier <devnexen@gmail.com>
David Carter <fresco.raja@gmail.com>
David Chase <drchase@google.com>
David Covert <davidhcovert@gmail.com>
David Crawshaw <david.crawshaw@zentus.com> <crawshaw@google.com> <crawshaw@golang.org>
David du Colombier <0intro@gmail.com>
David Finkel <david.finkel@gmail.com>
David Forsythe <dforsythe@gmail.com>
David G. Andersen <dave.andersen@gmail.com>
David Glasser <glasser@meteor.com>
David Golden <david@autopragmatic.com>
David Heuschmann <heuschmann.d@gmail.com>
David Howden <dhowden@gmail.com>
David Hubbard <dsh@google.com>
David Jakob Fritz <david.jakob.fritz@gmail.com>
David Jones <dxjones@gmail.com>
David Lazar <lazard@golang.org>
David Leon Gil <coruus@gmail.com>
David McLeish <davemc@google.com>
David Ndungu <dunjuguna@gmail.com>
David NewHamlet <david@newhamlet.com>
David Presotto <presotto@gmail.com>
David R. Jenni <david.r.jenni@gmail.com>
David Sansome <me@davidsansome.com>
David Stainton <dstainton415@gmail.com>
David Symonds <dssymonds@golang.org>
David Thomas <davidthomas426@gmail.com>
David Timm <dtimm@pivotal.io>
David Titarenco <david.titarenco@gmail.com>
David Tolpin <david.tolpin@gmail.com>
David Url <david@urld.io>
Frew Schmidt <github@frew.co>
Frits van Bommel <fbommenl@gmail.com>
G. Hussain Chinoı̈ <ghchinoiny@gmail.com>
Ga˚l Yahas <gaal@google.com>
Gabriel Arthr Ptu˚sson <gabriel@system.is>
Gabriel Aszalos <gabriel.aszalos@gmail.com>
Gabriel Guzman <gabe.guzman@gmail.com>
Gabriel Nelle <tehsphinx@web.de>
Gabriel Nicolas Avellaneda <avellaneda.gabriel@gmail.com>
Gabriel Rosenhouse <rosenhouse@gmail.com>
Gabriel Russell <gabriel.russell@gmail.com>
Gareth Paul Jones <gpj@foursquare.com>
Garret Kelly <gdk@google.com>
Gary Elliott <garyelliott@gmail.com>
Gaurish Sharma <contact@gaurishsharma.com>
Gautham Thambidorai <gautham.dorai@gmail.com>
Gauthier Jolly <gauthier.jolly@gmail.com>
Gawen Arab <gawen.arab@c.zen.ly>
Geert-Johan Riemer <gjr19912@gmail.com>
Genevieve Luyt <genevieve.luyt@gmail.com>
Gengliang Wang <lnwgl@gmail.com>
Geoff Berry <gberry.qdt@qualcommdatacentre.com>
Geoffroy Lorieux <lorieux.g@gmail.com>
Geon Kim <geon0250@gmail.com>
Georg Reinke <gueffey@gmail.com>
George Gkirtsou <ggkirtsou@gmail.com>
George Hartzell <hartzell@alerce.com>
George Shammas <george@shamm.as> <georgyo@gmail.com>
Gerasimos (Makis) Maropoulos <kataras2006@hotmail.com>
Gerasimos Dimitriadis <gedimitr@gmail.com>
Gergely Brautigam <skarlso777@gmail.com>
Gernot Vormayr <gvormayr@gmail.com>
Gert Cuykens <gert.cuykens@gmail.com>
Getulio Sanchez <valentin2507@gmail.com>
Ghazni Nattarshah <ghazni.nattarshah@gmail.com>
Gianguido Sora` <g.sora4@gmail.com>
Gideon Jan-Wessel Redelinghuys <gjredelinghuys@gmail.com>
Giles Lean <giles.lean@pobox.com>
Giovanni Bajo <rasky@develer.com>
GitHub User @aca (50316549) <acadx0@gmail.com>
GitHub User @ajnirp (1688456) <ajnirp@users.noreply.githubusercontent.com>
GitHub User @ajz01 (4744634) <ajzdenek@gmail.com>
GitHub User @alkesh26 (1019076) <alkesh26@gmail.com>
GitHub User @uropek (39370426) <uropek@gmail.com>
GitHub User @utkarsh-extc (53217283) <utkarsh.extc@gmail.com>
GitHub User @witchard (4994659) <witchard@hotmail.co.uk>
GitHub User @yah01 (12216890) <kagaminehuan@gmail.com>
GitHub User @yuanchh (1298735) <yuanchh12@gmail.com>
GitHub User @zikaeroh (48577114) <zikaeroh@gmail.com>
GitHub User @ZZMarquis (7624583) <zhonglingjian3821@163.com>
Giulio Iotti <dullgiulio@gmail.com>
Giulio Micheloni <giulio.micheloni@gmail.com>
Giuseppe Valente <gvalente@arista.com>
Gleb Stepanov <glebstepanov1992@gmail.com>
Glenn Brown <glennb@google.com>
Glenn Lewis <gmlewis@google.com>
Gordon Klaus <gordon.klaus@gmail.com>
Gordon Tyler <gordon@doxxx.net>
Graham King <graham4king@gmail.com>
Graham Miller <graham.miller@gmail.com>
Grant Griffiths <gpp493@gmail.com>
Greg Poirier <greg.istehbest@gmail.com>
Greg Steuck <gnezdo+github@google.com>
Greg Thelen <gtelen@bruck.com>
Greg Ward <greg@gerg.ca>
Grgoire Delattre <gregoire.delattre@gmail.com>
Gregory Man <man.gregory@gmail.com>
Gregory Petrosyan <gregory.petrosyan@gmail.com>
Guilherme Caruso <gui.martinscaruso@gmail.com>
Guilherme Garnier <guilherme.garnier@gmail.com>
Guilherme Goncalves <guilhermeaugustosg@gmail.com>
Guilherme Rezende <guilhermebr@gmail.com>
Guillaume J. Charmes <guillaume@charmes.net>
Gutherford Noack <gnoack@doxxx.net>
Guobiao Mei <meiguobiao@gmail.com>
Guoliang Wang <siamwgliang@gmail.com>
Gustav Paul <gustav.paul@gmail.com>
Gustav Westling <gustav@westling.xyz>
Gustavo Franco <gustavofranco@gmail.com>
Gustavo Niemeyer <gustavo@niemeyer.net> <n13m3y3r@gmail.com>
Gwenael Treguier <gwenn.kahz@gmail.com>
Gyu-Ho Lee <gyuhox@gmail.com>
H. brahim Gngr <sigungor@gmail.com>
Hajime Hoshi <hajimehoshi@gmail.com>
Hallgrímur Gunnarsson <halg@google.com>
HAMANO Tsukasa <hamano@osstech.co.jp>
Han-Wen Nienhuys <hanwen@doxxx.net>
Hang Qian <hangqian90@gmail.com>
Hanjun Kim <hallazzang@gmail.com>
Haosdent Huang <haosdent@gmail.com>
Harald Nordgren <haraldnordgren@gmail.com>
Ian Haken <ihaken@netflix.com>
Ian Kent <iankent85@gmail.com>
Ian Lance Taylor <iant@golang.org>
Ian Leue <ian@appboy.com>
Ian Zapolsky <ianzapolsky@gmail.com>
Ibrahim AshShohail <ibra.sho@gmail.com>
Icarus Sparry <golang@icarus.freeuk.com>
Iccha Sethi <icchasethi@gmail.com>
Idora Shinatose <idora.shinatose@gmail.com>
Ignacio Hagopian <jsign.uy@gmail.com>
Igor Bernstein <igor.bernstein@google.com>
Igor Dolzhikov <bluesriverz@gmail.com>
Igor Vashyst <ivashyst@gmail.com>
Igor Zhilianin <igor.zhilianin@gmail.com>
Ilya Yalovyy <yalovoy@gmail.com>
Ilya Sinelnikov <sidhmangh@gmail.com>
Ilya Tocar <ilya.tocar@intel.com>
INADA Naoki <songofacandy@gmail.com>
Inanc Gumus <m@inanc.io>
Ingo Gottwald <in.gottwald@gmail.com>
Ingo Krabbe <ikrabbe.ask@gmail.com>
Ingo Oeser <nightlyone@googlemail.com> <nightlyone@gmail.com>
Ioannis Georgoulas <geototti21@hotmail.com>
Irbe Krumina <irbekrm@gmail.com>
Irfan Sharif <irfanmahmoudsharif@gmail.com>
Irieda Noboru <irieda@gmail.com>
Isaac Ardis <isaac.ardis@gmail.com>
Isaac Wagner <ibw@isaacwagner.me>
Isfan Azhabil <isfanazhabil@gmail.com>
Iskander Sharipov <iskander.sharipov@intel.com> <quasilyte@gmail.com>
Issac Trolts <issactrots@google.com>
Ivan Babrou <ivan@cloudflare.com>
Ivan Bertona <ivan.berton@gmail.com>
Ivan Krasin <krasin@golang.org>
Ivan Kutuzov <arbrix@gmail.com>
Ivan Markin <sw@nogoeyst.net>
Ivan Moscoco <moscoso@gmail.com>
Ivan Osadchii <ivan.osadchii@gmail.com>
Ivan Sharavuev <shpiwan@gmail.com>
Ivan Trubach <mr.trubach@icloud.com>
Ivan Ukhov <ivan.ukhov@gmail.com>
Ivy Evans <ivyivyevans.net>
Jaana Burcu Dogan <jbd@google.com> <jbd@golang.org> <burcujdogan@gmail.com>
Jaap Aarts <jaap.aarts1@gmail.com>
Jack Britton <jacksbritton@gmail.com>
Jack Lindamood <jlindamo@justin.tv>
Jacob Baskin <jbaskin@gmail.com>
Jacob Blain Christen <dweomer5@gmail.com>
Lorenz Brun <lorenz@brun.one>
Lorenz Nickel <mail@lorenznickel.de>
Lorenzo Masini <rugginoso@develeer.com>
Lorenzo Stoakes <lstoakes@gmail.com>
Louis Kruger <louisk@google.com>
Luan Santos <cfluan@gmail.com>
Lubomir I. Ivanov <neolit123@gmail.com>
Luca Bruno <luca.bruno@coreos.com>
Luca Greco <luca.greco@alcacoop.it>
Lucas Bremgartner <lucas.bremgartner@gmail.com>
Lucas Clemente <lclemente@gmail.com>
Lucien Stuker <lucien.stuker@gmail.com>
Lucio De Re <lucio.dere@gmail.com>
Ludi Rehak <ludi317@gmail.com>
Luigi Riefolo <luigi.riefolo@gmail.com>
Luit van Drongelen <luitvd@gmail.com>
Luka Zakrajek <tr00.g33k@gmail.com>
Luka Zitnik <luka.zitnik@gmail.com>
Lukasz Milewski <lmmilewski@gmail.com>
Luke Champine <luke.champine@gmail.com>
Luke Curley <qpingu@gmail.com>
Luke Granger-Brown <git@lukegb.com>
Luke Young <bored-engineer@users.noreply.github.com>
Luna Duclos <luna.duclos@palmstonegames.com>
Luuk van Dijk <lvd@golang.org> <lvd@google.com>
Lyle Franklin <lylejfranklin@gmail.com>
Lynn Boger <laboger@linux.vnet.ibm.com>
Ma Peiqi <mapeiqi2017@gmail.com>
Maarten Bezemer <maarten.bezemer@gmail.com>
Maciej Dbski <maciejd@gmail.com>
Madhu Rajanna <madhupr007@gmail.com>
Magnus Hiie <magnus.hiie@gmail.com>
Maicon Costa <maiconscosta@gmail.com>
Mak Kolybabi <mak@kolybabi.com>
Maksym Trykur <maksym.trykur@gmail.com>
Mal Curtis <mal@mal.co.nz>
Manfred Touron <m@42.am>
Manigandan Dharmalingam <manigandan.jeff@gmail.com>
Manish Goregaokar <manishmail@gmail.com>
Manoj Dayaram <platform-dev@moovweb.com> <manoj.dayaram@moovweb.com>
Mansour Rahimi <rahimi.mnr@gmail.com>
Manu Garg <manugarg@gmail.com>
Manu S Ajith <neo@codingarena.in>
Manuel Mendez <mmendez534@gmail.com>
Marat Khabibullin <marat.khabibullin@jetbrains.com>
Marc Sanmiquel <marcsanmiquel@gmail.com>
Marc Weistroff <marc@weistroff.net>
Marc-Antoine Ruel <maruel@chromium.org>
Marcel Edmund Franke <marcel.edmund.franke@gmail.com>
Marcel van Lohuizen <mpvl@golang.org>
Marcelo Cantos <marcelo.cantos@gmail.com>
Marcelo E. Magallon <marcelo.maggallon@gmail.com>
Marco Hennings <marco.hennings@freiheit.com>
Marcus Weiner <marcus.weiner@gmail.com>
Marcus Willock <crazcalm@gmail.com>
Marga Manterola <marga@google.com>
Mariano Cano <mariano@smallstep.com>
Marin Bai <marin.basic02@gmail.com>
Mario Arranz <marioarranzr@gmail.com>
Marius A. Eriksen <marius@grailbio.com>
Marius Nuennerich <mnu@google.com>
Mark Adams <mark@markadams.me>
Mark Bucciarelli <mkbucci@gmail.com>
Mark Glines <mark@glines.org>
Mark Harrison <marhar@google.com>
Mark Percival <m@mdp.im>
Mark Pulford <mark@kyne.com.au>
Mark Rushakoff <mark.rushakoff@gmail.com>
Mark Ryan <mark.d.ryan@intel.com>
Mark Severson <miquella@gmail.com>
Mark Theunissen <mark.theunissen@gmail.com>
Mark Villacampa <m@markvillacampa.com>
Mark Wolfe <mark@wolfe.id.au>
Mark Zavislak <zavislak@google.com>
Marko Juhani Silokunnas <marko.silokunnas@gmail.com>
Marko Kevac <marko@kevac.org>
Marko Kungla <marko.kungla@gmail.com>
Marko Mikulicic <mkm@google.com>
Marko Mudrinic <mudrinic.mare@gmail.com>
Marko Tiikkaja <marko@joh.to>
Markus Duft <markus.duft@salomon.at>
Markus Sonderegger <marraison@gmail.com>
Markus Zimmermann <zimmski@gmail.com>
Marten Seemann <martenseemann@gmail.com>
Martin Asquino <martin.asquino@gmail.com>
Martin Bertschler <mbertschler@gmail.com>
Martin Garton <garton@gmail.com>
Martin Habbecke <marhab@google.com>
Martin Hamrle <martin.hamrle@gmail.com>
Martin Hoeffling <martin.hoeffling@gmx.de>
Martin Kreichgauer <martinkr@google.com>
Martin Kunc <martinkunc@users.noreply.github.com>
Martin Lindhe <martin.j.lindhe@gmail.com>
Martin Mohrman <moehrman@gmail.com> <martisch@uos.de>
Martin Neubauer <m.ne@gmx.net>
Martin Olsen <github.com@martinolsen.net>
Martin Olsson <martin@minimum.se>
Martin Probst <martin@probst.io>
Martin Sucha <anty.sk+git@gmail.com>
Martin Tournoij <martin@arp242.net>
Martins Sipenko <martins.sipenko@gmail.com>
Martynas Budinas <mabu@gmail.com>
Marvin Stenger <marvin.stenger94@gmail.com>
Marwan Sulaiman <marwan.sulaiman@work.co>
Maryan Hratson <gmarik@gmail.com>
Masahiro Furudate <masahiro.furudate@gmail.com>
Masahiro Wakame <vvakame@gmail.com>
Masaki Yoshida <yoshida.masaki@gmail.com>
Masaya Watanabe <sfbgwm30@gmail.com>
Mat Byczkowski <mbyczkowski@gmail.com>
Mat Ryer <thatmatryer@gmail.com>
Mt Gulya <mgulyas86@gmail.com>
Matej Bao <matejbaco@gmail.com>
Mateusz Amin <mateus.amin@gmail.com>
Mateusz Czapinski <czapkofan@gmail.com>
Mathias Beke <git@denbeke.be>
Mathias Hall-Andersen <mathias@hall-andersen.dk>
Mathias Leppich <mleppich@muhuq.de>
Mathieu Lonjaret <mathieu.lonjaret@gmail.com>
Mats Lidell <mats.lidell@cag.se> <mats.lidell@gmail.com>
Matt Aimonetti <mattaimonetti@gmail.com>
Matt Blair <me@matthewblair.net>
Matt Bostock <matt@mattbostock.com>
Matt Brown <mdbrown@google.com>
Matt Dee <mdee@hioscar.com>
Matt Drollette <matt@drollette.com>
Matt Harden <matt.harden@gmail.com>
Matt Jibson <matt.jibson@gmail.com>
Matt Joiner <anacrolix@gmail.com>
Matt Jones <mrjones@gmail.com>
Matt Juran <thepciet@gmail.com>
Matt Layher <mdlayher@gmail.com>
Matt Reiferson <mreiferson@gmail.com>
Matt Robenolt <matt@ydekproductions.com>
Matt Strong <mstrong1341@gmail.com>
Matt T. Proud <matt.proud@gmail.com>
Matt Williams <gh@mattyw.net> <mattyjwilliams@gmail.com>
Matthew Brennan <matty.brennan@gmail.com>
Matthew Broberg <mattbroberg@gmail.com>
Matthew Cottingham <mattcottingham@gmail.com>
Matthew Dempsky <mdempsky@google.com>
Matthew Denton <mdenton@skyportsystems.com>
Matthew Holt <Matthew.Holt+git@gmail.com>
Matthew Horssell <mattv.horsnell@gmail.com>
Michael Munday <mike.munday@ibm.com>
Michael Pearson <mipearson@gmail.com>
Michael Piatek <piatek@gmail.com>
Michael Pratt <mpratt@gmail.com>
Michael Schaller <michael@schaller.de>
Michael Schurter <michael.schurter@gmail.com>
Michael Shields <mshields@gmail.com>
Michael Stapelberg <michael@stapelberg.de> <mstplbrg@gmail.com>
Michael Steinert <mike.steinert@gmail.com>
Michael T. Jones <mtj@gmail.com> <michael.jones@gmail.com>
Michael Teichgraever <ntechgraever@gmx.de> <mt4swm@gmail.com>
Michael Traver <mtraver@gmail.com>
Michael Vetter <g.bluehut@gmail.com>
Michael Vogt <mvo@ubuntu.com>
Michal Bohuslav <mbohuslav@gmail.com>
Michal Cierniak <cierniak@gmail.com>
Micha Derkacz <ziutek@l.net.pl>
Michal Franc <lam.michal.franc@gmail.com>
Micha owicki <mlowicki@gmail.com>
Michal Pristas <michal.pristas@gmail.com>
Michal Rostecki <nrostecki@suse.de>
Michalis Kargakis <michaliskargakis@gmail.com>
Michel Lespinasse <swalken@gmail.com>
Mickael Kerjean <mickael.kerjean@gmail.com>
Mickey Reiss <mickeyreiss@gmail.com>
Mieken Gieben <miek@miek.nl> <remigius.gieben@gmail.com>
Miguel Acero <acero@gmail.com>
Miguel Mendez <stxmendez@gmail.com>
Miguel Molina <hi@mvader.me>
Mihai Borobocea <MihaiBorobocea@gmail.com>
Mihai Moldovan <ionic@ionic.de>
Mihai Todor <todormihai@gmail.com>
Mihail Minaev <minaev.mike@gmail.com>
Mikael Tillenius <mikti42@gmail.com>
Mike Andrews <mra@xoba.com>
Mike Appleby <mike@app.leby.org>
Mike Danese <mikedenise@gmail.com>
Mike Houston <mike@kothar.net>
Mike Kabischev <kabischev@gmail.com>
Mike Rosset <mike.rosset@gmail.com>
Mike Samuel <mikesamuel@gmail.com>
Mike Solomon <msolo@gmail.com>
Mike Strosaker <strosake@us.ibm.com>
Mike Tsao <mike@sowbug.com>
Mike Wiacek <mjiwacek@gmail.com>
Mikhail Fesenko <proggga@gmail.com>
Mikhail Gusarov <dottedmag@dottedmag.net>
Mikhail Panchenko <m@mihasya.com>
Serhii Aheienko <serhii.aheienko@gmail.com>
Seth Hoenig <seth.a.hoenig@gmail.com>
Seth Vargo <sethvargo@gmail.com>
Shahar Kohanim <skohanim@gmail.com>
Shamil Garatuev <garatuev@gmail.com>
Shane Hansen <shanemhansen@gmail.com>
Shang Jian Ding <sdjing3@ncsu.edu>
Shaozhen Ding <dsz0111@gmail.com>
Shaquille Wyan Que <shaqqwyan@gmail.com>
Shaun Dunning <shaun.dunning@uservoice.com>
Shawn Elliott <selliott@microsoft.com>
Shawn Ledbetter <sledbetter@google.com>
Shawn Smith <shawn.p.smith@gmail.com>
Shawn Walker-Salas <shawn.walker@oracle.com>
Shenghou Ma <minux@golang.org> <minux.ma@gmail.com>
Shengjing Zhu <zsj950618@gmail.com>
Shengyu Zhang <shengyu.zhang@chaint.com>
Shi Han Ng <shihanng@gmail.com>
Shijie Hao <haormj@gmail.com>
Shinji Tanaka <shinji.tanaka@gmail.com>
Shintaro Kaneko <kaneshin0120@gmail.com>
Shivakumar GN <shivakumar.gn@gmail.com>
Shivani Singhal <shivanisinghal2804@gmail.com>
Shivansh Rai <shivanshi@freebsd.org>
Shivashis Padhi <shivashispadhi@gmail.com>
Shubham Sharma <shubham.sha12@gmail.com>
Shun Fan <sfan@google.com>
Silvan Jegen <s.jegen@gmail.com>
Simarpreet Singh <simar@linux.com>
Simon Drake <simondrake1990@gmail.com>
Simon Ferquel <simon.ferquel@docker.com>
Simon Jefford <simon.jefford@gmail.com>
Simon Rawet <simon@rawet.se>
Simon Rozman <simon@rozman.si>
Simon Thulbourn <simon+github@thulbourn.com>
Simon Whitehead <chemnova@gmail.com>
Sina Siadat <siadat@gmail.com>
Sjoerd Siebinga <sjoerd.siebinga@gmail.com>
Sokolov Yura <funny.falcon@gmail.com>
Song Gao <song.gao.io>
Soojin Nam <jsunam@gmail.com>
Sren L. Hansen <soren@linux2go.dk>
Spencer Kocot <spencerkocot@gmail.com>
Spencer Nelson <s@spenczar.com>
Spencer Tung <spencertung@google.com>
Spring Mc <heresy.mc@gmail.com>
Srdjan Petrovic <spotifyic@google.com>
Sridhar Venkatakrishnan <sridhar@laddoo.net>
Srinidhi Kaushik <shrinidhi.kaushik@gmail.com>
StalkR <stalkr@stalker.net>
Stan Schwertly <stan@schwertly.com>
Stanislav Afanasev <sphp.progger@gmail.com>
Steeve Morin <steeve.morin@gmail.com>
Stefan Baehler <sbaehler@outbrain.com>
Stefan Nilsson <snilsson@nada.kth.se> <trollierjprofessor@gmail.com>
Stepan Shbalin <neverliberty@gmail.com>
Stephan Renatus <srenatus@chef.io>
Stephan Zuercher <zuercher@gmail.com>
Stphane Travostino <stephane.travostino@gmail.com>
Stephen Lewis <stephen@sock.org.uk>
Stephen Lu <steehs@users.noreply.github.com>
Stephen Ma <stephenn@golang.org>
Stephen McQuay <stephen@mcquay.me>
Stephen Searles <stephens2424@gmail.com>
Stephen Weinberg <stephen@q5comm.com>
Steve Francia <spf@golang.org>
Steve Gilbert <stevegilbert23@gmail.com>
Steve LoFurno <slofurno@gmail.com>
Steve McCoy <mccoyst@gmail.com>
Steve Mynott <steve.mynott@gmail.com>
Steve Newman <snewman@google.com>
Steve Phillips <elimisteve@gmail.com>
Steve Streetering <steve@stevestreeting.com>
Steven Buss <sbus@google.com>
Steven Elliot Harris <seharris@gmail.com>
Steven Erenst <stevenerenst@gmail.com>
Steven Hartland <steven.hartland@multiplay.co.uk>
Steven Littiebrant <imgroxx@gmail.com>
Steven Wilkin <stevenwilkin@gmail.com>
Stuart Jansen <sjansen@buscaluz.org>
Sue Spence <virtuallysue@gmail.com>
Sugu Sougoumarane <ssougou@gmail.com>
Suharsh Sivakumar <suharshs@google.com>
Sukrit Handa <sukrit.handa@utoronto.ca>
Sunny <me@darkowlzz.space>
Suriyaa Sundararuban <suriyaasundararuban@gmail.com>
Suyash <dextrous93@gmail.com>
Suzy Mueller <suzmuc@golang.org>
Sven Almgren <sven@tras.se>
Sven Blumenstein <svbl@google.com>
Sven Taute <sven.taut@gmail.com>
Sylvain Zimmer <sylvain@sylvainzimmer.com>
Syohei YOSHDIA <syohex@gmail.com>
Szabolcs Nagy <nsz@port70.net>
Taavi Kivisik <taavi.kivisik@gmail.com>
Tad Fisher <tadfisher@gmail.com>
Tad Glines <tad.glines@gmail.com>
Tadas Valiukas <tadovas@gmail.com>
Tadeo Kondrak <me@tadeo.ca>
Taesu Pyo <spyotaesu@gmail.com>
Tai Le <letientai299@gmail.com>
Taj Khattra <taj.khattra@gmail.com>
Takashi Matsuo <tmatsuo@google.com>
Takayoshi Nishida <takayoshi.nishida@gmail.com>
Takeshi YAMANASHI <9.nashi@gmail.com>
Takuto Ikuta <tikuta@google.com>
Takuya Ueda <uedatakuya@gmail.com>
Tai Shprecher <tshprecher@gmail.com>
Tams Gulcsi <tgulacsi78@gmail.com>
Tamir Duberstein <tamird@gmail.com>
Tao Qingyun <qingyunhua@gmail.com>
Tao Shen <shentaoaskying@gmail.com>
Tao Wang <twang2218@gmail.com>
Tarmigan Casebolt <tarmigan@gmail.com>
Taro Aoki <aiisu1230022@gmail.com>
Taru Karttunen <taruti@taruti.net>
Tatsuhiro Tsujikawa <tatsuhiro.t@gmail.com>
Tatsuya Kaneko <m.ddotx.f@gmail.com>
Taufiq Rahman <taufiqr88@gmail.com>
Teague Cole <tnc1443@gmail.com>
Ted Kornish <golang@tedkornish.com>
Tejasvi Nareddy <tejunareddy@gmail.com>
Terin Stock <terinjokes@gmail.com>
Terrel Shumway <gopher@shumway.us>
Tetsuo Kiso <tetsuokiso9@gmail.com>
Than McIntosh <thanm@google.com>
Thanabodee Charoenpiriyakij <wingyminus@gmail.com>
Thanatat Tamtac <acoshift@gmail.com>
The Hatsune Daishi <nao20010128@gmail.com>
Thiago Avelino <t@avelino.xxx>
Thiago Fransosi Farina <thiago.farina@gmail.com> <tfarina@chromium.org>
Thomas Alan Copeland <talan.copeland@gmail.com>
Thomas Bonfort <thomas.bonfort@gmail.com>
Thomas Bouldin <inlined@google.com>
Thomas Bruyelle <thomas.bruyelle@gmail.com>
Thomas Bushnell, BSG <tbushnell@google.com>
Thomas de Zeeuw <thomasdezeew@gmail.com>
Thomas Desrosiers <thomas.desr@gmail.com>
Thomas Habets <habets@google.com>
Thomas Kappler <t.kappler@gmail.com>
Thomas Meson <zllak@hycik.org>
Thomas Symborski <thomas.symborski@gmail.com>
Thomas Wanielista <tomwans@gmail.com>
Thorben Krueger <thorben.krueger@gmail.com>

Open Source Used In IOx-FogDirector-AC18 1.17.0 2874
Torben Schinke <torben.schinke@neotos.de>
Tormod Erevik Lea <tormodlea@gmail.com>
Toshihiro Shiino <shino.toshihiro@gmail.com>
Toshiki Shima <hayabusa1419@gmail.com>
Totoro W <tw19881113@gmail.com>
Travis Bischel <travis.bischel@gmail.com>
Travis Cline <travis.cline@gmail.com>
Trevor Strohman <trevor.strohman@gmail.com>
Trey Lawrence <lawrence.trey@gmail.com>
Trey Roessig <trey.roessig@gmail.com>
Trey Tacon <ttacon@gmail.com>
Tristan Amini <tamin01@ca.ibm.com>
Tristan Colgate <tcolgate@gmail.com>
Tristan Ooohry <ooohry@gmail.com>
Tristan Rice <rice@fn.ic>
Troels Thomsen <troels@thomsen.io>
Trung Nguyen <trung.n.k@gmail.com>
Tsuiji Daishiro <dram.dt.shonan@gmail.com>
Tudor Golubenco <tudor.g@gmail.com>
Tugdual Saunier <tugdual.saunier@gmail.com>
Tu Shao <sturbo89@gmail.com> <shantuo@google.com>
Tyler Bui-Palsulich <t palsulich@google.com>
Tyler Bunnell <tylerbunnell@gmail.com>
Tyler Treat <ttreat31@gmail.com>
Tyson Andre <tysonandre775@gmail.com>
Tzach Shabtay <tzachshabtay@gmail.com>
Tzu-Jung Lee <roylee17@currant.com>
Udalov Max <re.udalov@gmail.com> <shantuo@google.com>
Ugorji Nwoke <ugorji@gmail.com>
Ulf Holm Nielsen <doktor@dyregod.dk>
Ulrich Kunitz <uli.kunitz@gmail.com>
Umang Parmar <umangjparmar@gmail.com>
Uriel Mangado <uriel@berlinblue.org>
Urvi Patel <patelurvi38@gmail.com>
Uttam C Pawar <uttam.c.pawar@intel.com>
Vadim Grek <vadimprog@gmail.com>
Vadim Vygonets <unixdj@gmail.com>
Val Polouchkine <vpolouch@justin.tv>
Valentin Vidić <vvidic@valentin-vidic.from.hr>
Vee Zhang <veezhang@126.com> <veezhang@gmail.com>
Vega Garcia Luis Alfonso <vegacom@gmail.com>
Venil Noronha <venil@vmware.com>
Veselkov Konstantin <kostozyb@gmail.com>
Viacheslav Poturakov <vearutop@gmail.com>
Victor Chudnovsky <vchudnov@gmail.com>
Victor Vrančan <vranceanu+github@gmail.com>
Vignesh Ramachandra <vickyramachandra@gmail.com>
Vikas Kedia <vikask@google.com>
Yu Xuan Zhang <zyxsh@cn.ibm.com>
Yuichi Kishimoto <yk2220s@gmail.com>
Yuichi Nishiwaki <yuichi.nishiwiki@gmail.com>
Yuji Yaginuma <yuji.yaginuma@gmail.com>
Yuqi OKUSHI <huyuumi.dev@gmail.com>
Yuki Yugui Sonoda <yuguii@gmail.com>
Yuukihiro Nishinaka <6elpinal@gmail.com>
Yury Smolsky <yury@smolsky.by>
Yusuke Kagiwada <block.rxckin.beats@gmail.com>
Yuusei Kuwana <kuwana@kumama.org>
Yuval Pavel Zholkover <paulzhoi@gmail.com>
Yves Junqueira <yvesj@yvesj.com> <yves.junqueira@gmail.com>
Zac Bergquist <zbergquist99@gmail.com>
Zach Bintliff <zbintliff@gmail.com>
Zach Gershman <zach@thundertoken.com>
Zachary Amsden <zach@thundertoken.com>
Zachary Gershman <zgershman@pivotal.io>
Zak <zrzknill@gmail.com>
Zakatell Kanda <hi@zkanda.io>
Zellyn Hunter <zellyn@squareup.com> <zellyn@gmail.com>
Zev Goldstein <zev@gmail.com>
Zhang Boyang <zhangboyan@gmail.com>
Zheng Dayu <davidzheng23@gmail.com>
Zheng Xu <zheng.xu@arm.com>
Zhengyu He <hzy@gmail.com>
Zhongpeng Lin <zp@uber.com>
Zhongtao Chen <chenzhongtao@126.com>
Zhongwei Yao <zhongwei.yao@arm.com>
Zhou Peng <p@ctriple.cn>
Ziad Hatahet <hatahet@gmail.com>
Ziheng Liu <lzhfromustc@gmail.com>
Zorion Arrizabalaga <zorionk@gmail.com>
<a.maksadbek@gmail.com>
<max.faceless.frei@gmail.com>
<hagen1778@gmail.com>
<thunyk@gmail.com>
<faxriddinjon@gmail.com>
<zs349596@gmail.com>
<bronze1man@gmail.com>
Copyright 2010-2017 Mike Bostock
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the author nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

# This source code was written by the Go contributors.
# The master list of contributors is in the main Go distribution,
# visible at http://tip.golang.org/CONTRIBUTORS.
# People who have agreed to one of the CLAs and can contribute patches.
# The AUTHORS file lists the copyright holders; this file
# lists people. For example, Google employees are listed here
# but not in AUTHORS, because Google holds the copyright.
#
# https://developers.google.com/open-source/cla/individual
# https://developers.google.com/open-source/cla/corporate
#
# Names should be added to this file as:
#   Name <email address>
Raul Silvera <rsilvera@google.com>
Tipp Moseley <tipp@google.com>
Hyoun Kyu Cho <netforce@google.com>
Martin Spier <spiermar@gmail.com>
Taco de Wolff <tacodewolf@gmail.com>
Andrew Hunter <andrewhhunter@gmail.com>
# This source code was written by the Go contributors.
# The master list of contributors is in the main Go distribution,
# visible at https://tip.golang.org/CONTRIBUTORS.
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2019 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
Limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

Copyright 2009-2017 Andrea Leofreddi <a.leofreddi@vleo.net>. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall copyright holders or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of Andrea Leofreddi.

Copyright (c) 2015 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify, 
defend, and hold each Contributor harmless for any liability 
incurred by, or claims asserted against, such Contributor by reason 
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following 
boilerplate notice, with the fields enclosed by brackets "[]" 
replaced with your own identifying information. (Don't include 
the brackets!) The text should be enclosed in the appropriate 
comment syntax for the file format. We also recommend that a 
file or class name and description of purpose be included on the 
same "printed page" as the copyright notice for easier 
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); 
you may not use this file except in compliance with the License. 
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software 
distributed under the License is distributed on an "AS IS" BASIS, 
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. 
See the License for the specific language governing permissions and 
limitations under the License.

1.250 kubernetes-model 5.4.1
1.250.1 Available under license : 
No license file was found, but licenses were detected in source scan.

/**
 * Copyright (C) 2015 Red Hat, Inc.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
* /opt/cola/permits/1283432568_1646722587.58/0/kubernetes-model-admissionregistration-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/admission/v1beta1/AdmissionRequest.java
* /opt/cola/permits/1283432568_1646722587.58/0/kubernetes-model-admissionregistration-5-4-1-sources-jar/io/fabric8/kubernetes/api/model/admission/v1/AdmissionRequest.java

No license file was found, but licenses were detected in source scan.

* Copyright (C) 2015 Red Hat, Inc.
* Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*
1.251 hazelcast-client-protocol 1.8.0

1.251.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed.
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.252 base-files 10.1ubuntu2.11

1.252.1 Available under license:

Copyright (c) The Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF...
SUCH DAMAGE.

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if
you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link a program with the library, you must provide
complete object files to the recipients so that they can relink them
with the library, after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright
the library, and (2) offer you this license which gives you legal
permission to copy, distribute and/or modify the library.
Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The
former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.
In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.
Whether this is true is especially significant if the work can be
linked without the Library, or if the work is itself a library. The
threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data
structure layouts and accessors, and small macros and small inline
functions (ten lines or less in length), then the use of the object
file is unrestricted, regardless of whether it is legally a derivative
work. (Executables containing this object code plus portions of the
Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6.
Any executables containing that work also fall under Section 6,
whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or
link a “work that uses the Library” with the Library to produce a
work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer’s own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable “work that
uses the Library”, as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice
That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:
a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 1, February 1989

Copyright (C) 1989 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The license agreements of most software companies try to keep users at the mercy of those companies. By contrast, our General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. The General Public License applies to the Free Software Foundation's software and to any other program whose authors commit to using it. You can use it for your programs, too.

When we speak of free software, we are referring to freedom, not price. Specifically, the General Public License is designed to make sure that you have the freedom to give away or sell copies of free software, that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of a such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

The precise terms and conditions for copying, distribution and modification follow.

**GNU GENERAL PUBLIC LICENSE**

**TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION**

0. This License Agreement applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The “Program”, below, refers to any such program or work, and a "work based on the Program" means either the Program or any work containing the Program or a portion of it, either verbatim or with modifications. Each licensee is addressed as "you".

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this General Public License and to the absence of any warranty; and give any other recipients of the Program a copy of this General Public License along with the Program. You may charge a fee for the physical act of transferring a copy.

2. You may modify your copy or copies of the Program or any portion of it, and copy and distribute such modifications under the terms of Paragraph 1 above, provided that you also do the following:

   a) cause the modified files to carry prominent notices stating that you changed the files and the date of any change; and

   b) cause the whole of any work that you distribute or publish, that in whole or in part contains the Program or any part thereof, either with or without modifications, to be licensed at no charge to all third parties under the terms of this General Public License (except that you may choose to grant warranty protection to some or all third parties, at your option).

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the simplest and most usual way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these
conditions, and telling the user how to view a copy of this General Public License.

d) You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another independent work with the Program (or its derivative) on a volume of a storage or distribution medium does not bring the other work under the scope of these terms.

3. You may copy and distribute the Program (or a portion or derivative of it, under Paragraph 2) in object code or executable form under the terms of Paragraphs 1 and 2 above provided that you also do one of the following:

   a) accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Paragraphs 1 and 2 above; or,

   b) accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal charge for the cost of distribution) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Paragraphs 1 and 2 above; or,

   c) accompany it with the information you received as to where the corresponding source code may be obtained. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form alone.)

Source code for a work means the preferred form of the work for making modifications to it. For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs, or for standard header files or definitions files that accompany that operating system.

4. You may not copy, modify, sublicense, distribute or transfer the Program except as expressly provided under this General Public License. Any attempt otherwise to copy, modify, sublicense, distribute or transfer the Program is void, and will automatically terminate your rights to use the Program under this License. However, parties who have received copies, or rights to use copies, from you under this General Public License will not have their licenses terminated so long as such parties remain in full compliance.

5. By copying, distributing or modifying the Program (or any work based
on the Program) you indicate your acceptance of this license to do so, and all its terms and conditions.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

7. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of the license which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the license, you may choose any version ever published by the Free Software Foundation.

8. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

9. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

10. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to humanity, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) 19yy  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19xx name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the
commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (a program to direct compilers to make passes at assemblers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

That's all there is to it!

The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.
"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.
The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do
not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

   b) The work must carry prominent notices stating that it is
released under this License and any conditions added under section
7. This requirement modifies the requirement in section 4 to
"keep intact all notices".

c) You must license the entire work, as a whole, under this
License to anyone who comes into possession of a copy. This
License will therefore apply, along with any applicable section 7
additional terms, to the whole of the work, and all its parts,
regardless of how they are packaged. This License gives no
permission to license the work in any other way, but it does not
invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display
Appropriate Legal Notices; however, if the Program has interactive
interfaces that do not display Appropriate Legal Notices, your
work need not make them do so.

A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product
   (including a physical distribution medium), accompanied by the
   Corresponding Source fixed on a durable physical medium
   customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product
   (including a physical distribution medium), accompanied by a
   written offer, valid for at least three years and valid for as
   long as you offer spare parts or customer support for that product
   model, to give anyone who possesses the object code either (1) a
   copy of the Corresponding Source for all the software in the
   product that is covered by this License, on a durable physical
   medium customarily used for software interchange, for a price no
   more than your reasonable cost of physically performing this
conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must
suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly
provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever
licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner...
consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.
Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.
IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.
Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes
of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.

3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

   b) use the modified Package only within your corporation or organization.

   c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

   d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or
executable form, provided that you do at least ONE of the following:

a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

b) accompany the distribution with the machine-readable source of the Package with your modifications.

c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is,
when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.
The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.


The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombinining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this
License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

1.253 vertx-hazelcast 3.9.4

1.253.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.254 mawk 1.3.3 17ubuntu3

1.254.1 Available under license:

Upstream source http://invisible-island.net/mawk/mawk.html

Current maintainer: Thomas Dickey <dickey@invisible-island.net>

-------------------------------------------------------------------------------
mawk 1.3.4 and updates, Copyright 2008-2019 by Thomas E. Dickey
mawk 1.3.4 includes substantial work by others:
Copyright 2009-2010 by Jonathan Nieder
Copyright 2005 by Aleksey Cheusov
mawk 1.3.3 Nov 1996, Copyright (C) Michael D. Brennan

Mawk is distributed without warranty under the terms of
the GNU General Public License, version 2, 1991.

-------------------------------------------------------------------------------

Files: aclocal.m4
Licence: other-BSD
Copyright: 2008-2018,2019 by Thomas E. Dickey

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, distribute with modifications, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright
holders shall not be used in advertising or otherwise to promote the
sale, use or other dealings in this Software without prior written
authorization.
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN
AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNEC-
TION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not
be used in advertising or otherwise to promote the sale, use or other deal-
ings in this Software without prior written authorization from the X Consor-
tium.

FSF changes to this file are in the public domain.

Calling this script install-sh is preferred over install.sh, to prevent
`make' implicit rules from creating a file called install from it
when there is no Makefile.

This script is compatible with the BSD install script, but was written
from scratch. It can only install one file at a time, a restriction
shared with many OS’s install programs.

Copyright: 2012-2019 Thomas E. Dickey
Licence: other-BSD
Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of the above listed
copyright holder(s) not be used in advertising or publicity pertaining
to distribution of the software without specific, written prior
permission.
THE ABOVE LISTED COPYRIGHT HOLDER(S) DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL THE ABOVE LISTED COPYRIGHT HOLDER(S) BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2'

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.255 jetty 9.4.31.v20200723

1.255.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.
“Contributor” means any person or entity that distributes the Program.

“Licensed Patents” mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

“Program” means the Contributions distributed in accordance with this Agreement.

“Recipient” means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient’s responsibility to acquire that license before distributing the Program.
   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under
its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the
Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself
(excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd
unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

Oracle OpenJDK
If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.  
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

------

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

------

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.256 vertx-web-common 3.9.4
1.256.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2011-2018 The original author or authors
 * ------------------------------------------------------
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 * *
 * The Eclipse Public License is available at
 * *
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 * *
 * You may elect to redistribute this code under either of these licenses.
 */

Found in path(s):
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/ext/web/multipart/impl/FormDataPartImpl.java
/*
 * Copyright 2018 Red Hat, Inc.
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 * The Eclipse Public License is available at
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 * You may elect to redistribute this code under either of these licenses.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright (c) 2011-2013 The original author or authors
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 * The Eclipse Public License is available at
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 * You may elect to redistribute this code under either of these licenses.
 */

No license file was found, but licenses were detected in source scan.
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/codec/impl/BodyCodecImpl.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 Red Hat, Inc.
 * 
 * All rights reserved. This program and the accompanying materials
 * are made available under the terms of the Eclipse Public License v1.0
 * and Apache License v2.0 which accompanies this distribution.
 * 
 * The Eclipse Public License is available at
 * 
 * The Apache License v2.0 is available at
 * http://www.opensource.org/licenses/apache2.0.php
 * 
 * You may elect to redistribute this code under either of these licenses.
 */

Found in path(s):
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/codec/impl/StreamingBodyCodec.java
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/common/template/CachingTemplateEngine.java
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/common/template/BodyStream.java
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/common/template/TemplateEngine.java
* /opt/cola/permits/1101064227_1603114228.6/0/vertx-web-common-3-9-4-sources-jar/io/vertx/extension/web/common/template/impl/ConcurrentLRUCache.java

1.257 snake-yaml 1.27

1.257.1 Available under license:

Found license 'GNU Lesser General Public License' in ' This module is multi-licensed and may be used under the terms EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html AL, Apache License, V2.0 or later, http://www.apache.org/licenses BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

Found license 'Eclipse Public License 1.0' in ' This module is multi-licensed and may be used under the terms EPL, Eclipse Public License, V1.0 or later, http://www.eclipse.org/legal LGPL, GNU Lesser General Public License, V2.1 or later, http://www.gnu.org/licenses/lgpl.html GPL, GNU General Public License, V2 or later, http://www.gnu.org/licenses/gpl.html AL, Apache License, V2.0 or later, http://www.apache.org/licenses BSD, BSD License, http://www.opensource.org/licenses/bsd-license.php * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'
1.258 c3p0 0.9.5.4
1.258.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'This library is free software; you can redistribute it and/or modify 1)
The GNU Lesser General Public License (LGPL), version 2.1, as published by the Free Software Foundation 2) The
Eclipse Public License (EPL), version 1.0 This software is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE. LGPL v2.1: http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html'

Found license 'GNU Lesser General Public License' in 'This library is free software; you can redistribute it and/or
modify 1) The GNU Lesser General Public License (LGPL), version 2.1, as published by the Free Software
Foundation 2) The Eclipse Public License (EPL), version 1.0 This software is distributed in the hope that it will be
useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. LGPL v2.1: http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html'

1.259 libnlroute 3.2.29-0ubuntu3
1.259.1 Available under license:

This package was debianized by Heiko Stuebner <heiko@sntech.de> on
Tue, 16 Jan 2006 12:00:46 +0000.
The packaging is based on Michael Biebl's <biebl@teco.edu> original packaging
of libnl1.

It was downloaded from https://github.com/thom311/libnl/releases

Upstream Author:

Thomas Graf <tgraf@suug.ch>

Copyright:

lib/route/addr.c
include/netlink/route/addr.h

Copyright (c) Thomas Graf <tgraf@suug.ch>
Baruch Even <baruch@ev-en.org>

lib/route/cls/u32.c
lib/route/cls/fw.c
lib/route/sch/htb.c
include/netlink/route/cls/fw.h
include/netlink/route/sch/htb.h

Copyright (c) Thomas Graf <tgraf@suug.ch>
Copyright (c) Petr Gotthard <petr.gotthard@siemens.com>
Copyright (c) Siemens AG Oesterreich

lib/netfilter/log_msg.c
lib/netfilter/ct.c
include/netlink/netfilter/log_msg.h
include/netlink/netfilter/log.h
lib/netfilter/log_obj.c

Copyright (c) Thomas Graf <tgraf@suug.ch>
Copyright (c) Philip Craig <philipc@snapgear.com>
Copyright (c) Patrick McHardy <kaber@trash.net>
Copyright (c) Secure Computing Corporation

include/netlink/netfilter/queue_msg.h
lib/netfilter/queue_msg_obj.c
lib/netfilter/queue_msg.c
lib/netfilter/queue.c
lib/netfilter/netfilter.c
lib/netfilter/queue_obj.c
include/netlink/netfilter/netfilter.h
include/netlink/netfilter/queue.h
src/nf-queue.c

Copyright (c) Patrick McHardy <kaber@trash.net>

include/netlink/xfrm/selector.h
include/netlink/xfrm/sa.h
include/netlink/xfrm/ae.h
include/netlink/xfrm/sp.h
include/netlink/xfrm/template.h
include/netlink/xfrm/lifetime.h
lib/xfrm/sa.c
lib/xfrm/template.c
lib/xfrm/ae.c
lib/xfrm/sp.c
lib/xfrm/selector.c
lib/xfrm/lifetime.c

Copyright (C) 2012 Texas Instruments Incorporated - http://www.ti.com/

All other *.c and *.h files not mentioned above are copyright of:
Copyright (c) 2003-2006 Thomas Graf <tgraf@suug.ch>

License:

src/nl-addr-add.c
src/nl-addr-list.c
src/nl-cls-add.c
src/cls/utils.c
src/cls/cgroup.c
src/cls/utils.h
src/cls/basic.c
src/nl-addr-delete.c:

This library is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation version 2 of the License.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in /usr/share/common-licenses/GPL-2

include/netlink/xfrm/selector.h
include/netlink/xfrm/sa.h
include/netlink/xfrm/ae.h
include/netlink/xfrm/sp.h
include/netlink/xfrm/template.h
include/netlink/xfrm/lifetime.h
lib/xfrm/sa.c
lib/xfrm/template.c
lib/xfrm/ae.c
lib/xfrm/sp.c
lib/xfrm/selector.c
lib/xfrm/lifetime.c

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Texas Instruments Incorporated nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

All other *.c and *.h files not mentioned above:

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation version 2.1 of the License.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

On Debian GNU/Linux systems, the complete text of the GNU Lesser General Public License can be found in /usr/share/common-licenses/LGPL-2.1

1.260 zstd 1.3.3+dfsg 2ubuntu1.2

1.260.1 Available under license:

BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

* Neither the name Facebook nor the names of its contributors may be used to
  endorse or promote products derived from this software without specific
  prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where this notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: Zstd
Source: https://github.com/facebook/zstd
Files-Excluded: appveyor.yml
circle.yml
build/*
programs/windres/*
.travis.yml
.buckversion
.buckconfig

Files: *
Copyright: 2013-2016, Yann Collet
2016, Przemyslaw Skibinski
2016, Facebook, Inc.
License: BSD-3-clause-with-patent-grant

Files: zlibWrapper/examples/*.c
Copyright: 1995-2006, 2011 Jean-loup Gailly
License: zlib

Files: zlibWrapper/gz*.c
License: zlib

License: zlib
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgement in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Files: contrib/linux-kernel/fs/squashfs/*
Copyright: 2017 Facebook
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

On Debian systems you can find a full copy of the GNU General Public License version 2 at /usr/share/common-licenses/GPL-2.

Files: lib/dictBuilder/divsufsort.*
Copyright: 2003-2008, Yuta Mori
License: Expat

Files: examples/*
Copyright: 2016-present, Yann Collet, Facebook, Inc.
License: BSD-3-clause-with-patent-grant and GPL-2

Files: debian/*
Copyright: 2015-2016 Kevin Murray <spam@kdmurray.id.au>
License: Expat

License: Expat
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

License: GPL-2
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License, v2, as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2'.

License: BSD-3-clause-with-patent-grant
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.


* Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Comment: A generic additional grant of patent rights is made by Facebook to the users of Zstd, and is reproduced below (from ./PATENTS):

**Additional Grant of Patent Rights Version 2**

"Software" means the Zstandard software distributed by Facebook, Inc.

Facebook, Inc. ("Facebook") hereby grants to each recipient of the Software ("you") a perpetual, worldwide, royalty-free, non-exclusive, irrevocable (subject to the termination provision below) license under any Necessary Claims, to make, have made, use, sell, offer to sell, import, and otherwise transfer the Software. For avoidance of doubt, no license is granted under Facebook’s rights in any patent claims that are infringed by (i) modifications to the Software made by you or any third party or (ii) the Software in combination with any software or other technology.

The license granted hereunder will terminate, automatically and without notice, if you (or any of your subsidiaries, corporate affiliates or agents) initiate directly or indirectly, or take a direct financial interest in, any Patent Assertion: (i) against Facebook or any of its subsidiaries or corporate affiliates, (ii) against any party if such Patent Assertion arises in whole or in part from any software, technology, product or service of Facebook or any of its subsidiaries or corporate affiliates, or (iii) against any party relating to the Software. Notwithstanding the foregoing, if Facebook or any of its subsidiaries or corporate affiliates files a lawsuit alleging patent infringement against you in the first instance, and you respond by filing a patent infringement counterclaim in that lawsuit against that party that is unrelated to the Software, the license granted hereunder will not terminate under section (i) of this paragraph due to such counterclaim.

A "Necessary Claim" is a claim of a patent owned by Facebook that is necessarily infringed by the Software standing alone.

A "Patent Assertion" is any lawsuit or other action alleging direct, indirect,
or contributory infringement or inducement to infringe any patent, including a cross-claim or counterclaim.

1.261 librados 12.2.13-0ubuntu0.18.04.10

1.261.1 Available under license:

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: ceph
Upstream-Contact: Sage Weil <sage@newdream.net>
Source: http://ceph.com/
Files-Excluded:
debian
src/civetweb/examples/websocket_client/ssl/server.key.orig
src/civetweb/resources/cert/client.key.orig
src/civetweb/resources/cert/server.key.orig
src/erasure-code/jerasure/jerasure/Examples/makefile.orig
src/erasure-code/jerasure/jerasure/include/config.h.in-
src/erasure-code/jerasure/jerasure/makefile.orig

Files: *
Copyright: 2004-2014 Sage Weil <sage@newdream.net>
2004-2014 Inktank <info@inktank.com>
    Inktank, Inc
    Inktank Storage, Inc.
2012-2014 Red Hat <contact@redhat.com>
2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
2013 CohortFS, LLC
2004-2011 Dreamhost
2013 eNovance SAS <licensing@enovance.com>
2014 Adam Crume <adamcrume@gmail.com>
2012 Florian Haas, hastexo
2010 Greg Farnum <gregf@hq.newdream.net>
2014 John Spray <john.spray@inktank.com
2004-2012 New Dream Network
2014 Sebastien Ponce <sebastien.ponce@cern.ch>
2011 Stanislav Sedov <stas@FreeBSD.org>
2013-2014 UnitedStack <haomai@unitedstack.com>
2011 Wido den Hollander <wido@widodh.nl>
License: LGPL-2.1

Files: src/erasure-code/jerasure/ErasureCode*
src/erasure-code/ErasureCode*
src/erasure-code/isa/*
src/include/str_map.h
src/test/common/test_str_map.cc
src/test/erasure-code/*
src/test/rgw/test_rgw_manifest.cc
Copyright: 2014 CERN/Switzerland
2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
2014 Red Hat <contact@redhat.com>
2013 eNovance SAS <licensing@enovance.com>
License: LGPL-2.1+

Files: src/erasure-code/isa/isa-l/erasure_code/*
Copyright: 2011-2014 Intel Corporation
License: BSD-3-clause

Files: src/rocksdb/*
Copyright: 2004-2013 Facebook, Inc.
2011 The LevelDB Authors
2009 Google Inc.
License: BSD-3-clause
Comment:
Additional Grant of Patent Rights

Software means the rocksdb software distributed by Facebook, Inc.

Facebook hereby grants you a perpetual, worldwide, royalty-free, non-exclusive, irrevocable (subject to the termination provision below) license under any rights in any patent claims owned by Facebook, to make, have made, use, sell, offer to sell, import, and otherwise transfer the Software. For avoidance of doubt, no license is granted under Facebooks rights in any patent claims that are infringed by (i) modifications to the Software made by you or a third party, or (ii) the Software in combination with any software or other technology provided by you or a third party.

The license granted hereunder will terminate, automatically and without notice, for anyone that makes any claim (including by filing any lawsuit, assertion or other action) alleging (a) direct, indirect, or contributory infringement or inducement to infringe any patent: (i) by Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, (ii) by any party if such claim arises in whole or in part from any software, product or service of Facebook or any of its subsidiaries or affiliates, whether or not such claim is related to the Software, or (iii) by any party relating to the Software; or (b) that any right in any patent claim of Facebook is invalid or unenforceable.

Files: src/rocksdb/util/xxhash.*
Copyright: 2012-2014, Yann Collet.
License: BSD-2-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: src/mount/canonicalize.c
       src/test/common/test_config.cc
       src/test/crush/TestCrushWrapper.cc
       src/test/common/Throttle.cc
       src/test/objectstore/chain_xattr.cc
       src/test/mon/mon-test-helpers.sh
       src/test/objectstore/chain_xattr.cc
       src/test/osd/osd-test-helpers.sh
       src/ceph-disk
       src/stop.sh
Copyright: 1993      Rick Sladkey <jrs@world.std.com>
       2013      Inktank <info@inktank.com>
       2013-2014 Cloudwatt <libre.licensing@cloudwatt.com>
License: LGPL-2+

Files: src/os/btrfs_ioctl.h
       src/test/mon/PGMap.cc
Copyright: 2007 Oracle. All rights reserved.
       2014 Inktank <info@inktank.com>
License: GPL-2

Files: src/common/ceph_hash.cc
License: public-domain
This file uses Robert Jenkin's hash function as detailed at:

   http://burtleburtle.net/bob/hash/evahash.html

This is in the public domain.
Files: src/common/bloom_filter.hpp
Copyright: 2000 Arash Partow
License: Boost-Software-License-1.0
Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Files: src/common/crc32c_intel*
Copyright: 2012-2013 Intel Corporation All Rights Reserved.
License: BSD-3-clause

Files: src/common/sctp_crc32.c
Copyright: 2001-2007, by Cisco Systems, Inc. All rights reserved, 2004-2006 Intel Corporation - All Rights Reserved
License: BSD-3-clause

Files: src/erasure-code/jerasure/gf-complete/*/*
Copyright: 2013 James S. Plank
Ethan L. Miller
Kevin M. Greenan
Benjamin A. Arnold
John A. Burnum
Adam W. Disney
Allen C. McBride
License: BSD-3-clause
Comment:
https://bitbucket.org/jimplank/gf-complete

Files: src/erasure-code/jerasure/jerasure/*/*
Copyright: 2011-2013 James S. Plank <plank@cs.utk.edu>
FOR A PARTICULAR PURPOSE. See the GNU General Public License for more
details.

The complete text of the GNU General Public License version 3
can be found in '/usr/share/common-licenses/GPL-3' file.

Files: src/mount/mtab.c
Copyright: util-linux-ng AUTHORS
License: GPL-2+
Comment:
"mount/fstab.c" from line 559:
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/mount-deprecated/fstab.c?h=v2.22#n559
https://git.kernel.org/cgit/utils/util-linux/util-linux.git/tree/README.licensing

Files: src/test/librbd/fsx.c
Copyright: 1991, NeXT Computer, Inc.
License: APSL-2.0
The contents of this file constitute Original Code as defined in and
are subject to the Apple Public Source License Version 2.0 (the
"License"). You may not use this file except in compliance with the
License. Please obtain a copy of the License at

This Original Code and all software distributed under the License are
distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. Please see the
License for the specific language governing rights and limitations
under the License.
Comment:
http://codemonkey.org.uk/projects/fsx/
http://codemonkey.org.uk/projects/fsx/fsx-macosforge/fsx.c

Files: man/*
    debian/man/*
Copyright: 2010-2014, Inktank Storage, Inc. and contributors.
License: CC-BY-SA-3.0

Files: debian/*
Copyright: 2010 Sage Weil <sage@newdream.net>
    2010 Canonical, Ltd.
    2011-2013 Lszl Bszrmnyi (GCS) <gcs@debian.org>
    2013-2014 James Page <james.page@ubuntu.com>
    2014 Dmitry Smirnov <onlyjob@debian.org>
License: LGPL-2.1

License: GPL-2
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2' file.

License: GPL-2+
This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU General Public License version 2 can be found in `/usr/share/common-licenses/GPL-2'.

License: LGPL-2.1
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 2.1 as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

License: LGPL-2.1+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU Lesser General Public License can be found in `/usr/share/common-licenses/LGPL-2.1`.

License: LGPL-2+
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 2 (or later) as published by the Free Software Foundation.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the complete text of the GNU Lesser General Public License 2 can be found in `/usr/share/common-licenses/LGPL-2`.

License: Expat
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Comment:
This license also known as "MIT" however FSF consider "MIT" labelling ambiguous and copyright-format specification recommend to label such license as "Expat".

License: CC-BY-SA-3.0
Creative Commons Attribution-ShareAlike 3.0 Unported
moving image ("synching") will be considered an Adaptation for the purpose of this License.

b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.

c. "Creative Commons Compatible License" means a license that is listed at http://creativecommons.org/compatiblelicenses that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.

d. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.

e. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.

f. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

g. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

h. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the
literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

i. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

j. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

k. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;

b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked “The original work was translated from English to Spanish,” or a modification could indicate “The original work has been modified.”;

c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,

d. to Distribute and Publicly Perform Adaptations.

e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;

ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,

iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You
Distribute or Publicly Perform. You may not offer or impose any terms on
the Work that restrict the terms of this License or the ability of the
recipient of the Work to exercise the rights granted to that recipient
under the terms of the License. You may not sublicense the Work. You
must keep intact all notices that refer to this License and to the
disclaimer of warranties with every copy of the Work You Distribute or
Publicly Perform. When You Distribute or Publicly Perform the Work, You
may not impose any effective technological measures on the Work that
restrict the ability of a recipient of the Work from You to exercise the
rights granted to that recipient under the terms of the License. This
Section 4(a) applies to the Work as incorporated in a Collection, but
this does not require the Collection apart from the Work itself to be
made subject to the terms of this License. If You create a Collection,
upon notice from any Licensor You must, to the extent practicable,
remove from the Collection any credit as required by Section 4(c), as
requested. If You create an Adaptation, upon notice from any Licensor
You must, to the extent practicable, remove from the Adaptation any
credit as required by Section 4(c), as requested.

b. You may Distribute or Publicly Perform an Adaptation only under the
terms of: (i) this License; (ii) a later version of this License with
the same License Elements as this License; (iii) a Creative Commons
jurisdiction license (either this or a later license version) that
contains the same License Elements as this License (e.g.,
Attribution-ShareAlike 3.0 US); (iv) a Creative Commons Compatible
License. If you license the Adaptation under one of the licenses
mentioned in (iv), you must comply with the terms of that license. If
you license the Adaptation under the terms of any of the licenses
mentioned in (i), (ii) or (iii) (the "Applicable License"), you must
comply with the terms of the Applicable License generally and the
following provisions: (I) You must include a copy of, or the URI for,
the Applicable License with every copy of each Adaptation You Distribute
or Publicly Perform; (II) You may not offer or impose any terms on the
Adaptation that restrict the terms of the Applicable License or the
ability of the recipient of the Adaptation to exercise the rights
granted to that recipient under the terms of the Applicable License;
(III) You must keep intact all notices that refer to the Applicable
License and to the disclaimer of warranties with every copy of the Work
as included in the Adaptation You Distribute or Publicly Perform; (IV)
when You Distribute or Publicly Perform the Adaptation, You may not
impose any effective technological measures on the Adaptation that
restrict the ability of a recipient of the Adaptation from You to
exercise the rights granted to that recipient under the terms of the
Applicable License. This Section 4(b) applies to the Adaptation as
incorporated in a Collection, but this does not require the Collection
apart from the Adaptation itself to be made subject to the terms of the
Applicable License.
c. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv), consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

d. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer
UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of
the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with
Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License.

Creative Commons may be contacted at http://creativecommons.org/.

License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.262 asm 3.3.1

1.262.1 Available under license:
No license file was found, but licenses were detected in source scan.

/***
 * ASM: a very small and fast Java bytecode manipulation framework
 * Copyright (c) 2000-2007 INRIA, France Telecom
***/
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/MethodWriter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/tree/analysis/Value.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/tree/LabelNode.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/FieldWriter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/util/CheckSignatureAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/util/CheckClassAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/tree/analysis/BasicValue.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/tree/MethodInsnNode.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/AnnotationVisitor.java
No license file was found, but licenses were detected in source scan.

ASM XML Adapter
Copyright (c) 2004, Eugene Kuleshov
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/XML.dtd
No license file was found, but licenses were detected in source scan.

/***/
* ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-
  jar/org/objectweb/asm/commons/InstructionAdapter.java
No license file was found, but licenses were detected in source scan.

* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-
  jar/org/objectweb/asm/xml/package.html

No license file was found, but licenses were detected in source scan.

/***
* ASM XML Adapter
* Copyright (c) 2004, Eugene Kuleshov
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.
*/
Found in path(s):
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/SAXFieldAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/SAXAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/SAXCodeAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/SAXClassAdapter.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/ASMContentHandler.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/Processor.java
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4.jar/org/objectweb/asm/xml/SAXAnnotationAdapter.java

No license file was found, but licenses were detected in source scan.

2005 INRIA, France Telecom
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
*    contributors may be used to endorse or promote products derived from
*    this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-jar/org/objectweb/asm/package.html
* /opt/cola/permits/1128777990_1612054951.89/0/asm-all-3-3-1-sources-4-
ASM: a very small and fast Java bytecode manipulation framework
* Copyright (c) 2000-2007 INRIA, France Telecom
* All rights reserved.
* * Redistribution and use in source and binary forms, with or without
  * modification, are permitted provided that the following conditions
  * are met:
  * 1. Redistributions of source code must retain the above copyright
     * notice, this list of conditions and the following disclaimer.
  * 2. Redistributions in binary form must reproduce the above copyright
     * notice, this list of conditions and the following disclaimer in the
     * documentation and/or other materials provided with the distribution.
  * 3. Neither the name of the copyright holders nor the names of its
     * contributors may be used to endorse or promote products derived from
     * this software without specific prior written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
  * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
  * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
  * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
  * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
  * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
  * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
  * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
  * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
  * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
  * THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):
* /opt/ cola/ permits/ 1128777990_1612054951. 89/0/asm-all-3-3-1-sources-4-jar/org/ objectweb/asm/util/Traceable.java
* /opt/ cola/ permits/ 1128777990_1612054951. 89/0/asm-all-3-3-1-sources-4-jar/org/ objectweb/asm/util/ASMifiable.java
1.263 procps 3.3.12-3ubuntu1.2

1.263.1 Available under license:

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: procps-ng
Source: https://gitlab.com/procps-ng/procps

Files: *
Copyright: 1998-2004 Albert Cahalan
1991 Tony Rems <rembo@unisoft.com>
1993 Larry Greenfield
1996 Charles Blake
1999 Mike Coleman <mkc@acm.org>
2004 Nicholas Mieg
2003 Chris Rivera
2003 Robert Love <rml@tech9.net>
2008 Morty Abzug
2009 Jarrod Lowe <procps@rod.net>
1992 Branko Lankester
???? David Engel <david@ods.com>
1992-1998 Michael K. Johnson <johnsonm@sunsite.unc.edu>
2012 Craig Small <csmall@enc.com.au>
1995 Martin Schulze <joey@infodrom.north.de>
License: LGPL-2.1+

Files: top/*
Copyright: 2002-2014 James C. Warner
License: LGPL-2.0+

Files: pgrep.*
Copyright: 2000 Kjetil Torgrim Homme <kjetilho@ifi.uio.no>
2002,2006 Albert Cahalan
2012 Roberto Polli <rpolli@babel.it>
License: GPL-2.0+

Files: pidof.*
Copyright: 2013 Jaromir Capik <jcapik@redhat.com>
License: GPL-2.0+

Files: free.*
Copyright: 2011 Sami Kerola <kerolasa@iki.fi>
2002-2003 Robert Love <rml@tech9.net>
1992 Brian Edmonds
1992 Rafał Maszkowski
2004 Albert Cahalan
License: GPL-2.0+
Files: sysctl.*
Copyright: 1999 George Staikos
License: GPL-2.0+

Files: debian/*/ 
Copyright: 1997-2015 Craig Small <csmall@debian.org>
1996-1197 Helmut Geyer <Helmut.Geyer@iwr.uni-heidelberg.de>
License: GPL-2.0+

License: GPL-2.0+
This package is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
.
This package is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
.
You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>
.
On Debian systems, the complete text of the GNU General
Public License version 2 can be found in "/usr/share/common-licenses/GPL-2".

License: LGPL-2.0+
This package is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.
.
This package is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.
.
You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.
.
On Debian systems, the complete text of the GNU Lesser General
Public License can be found in "/usr/share/common-licenses/LGPL-2".

License: LGPL-2.1+
This package is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This package is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

On Debian systems, the complete text of the GNU Lesser General Public License can be found in "/usr/share/common-licenses/LGPL-2.1".

1.264 postgresql-jdbc 9.3-1103-jdbc41

1.264.1 Available under license:

<OWNER> = Regents of the University of California
<ORGANIZATION> = University of California, Berkeley
<YEAR> = 1998

In the original BSD license, both occurrences of the phrase "COPYRIGHT HOLDERS AND CONTRIBUTORS" in the disclaimer read "REGENTS AND CONTRIBUTORS".

Here is the license template:

Copyright (c) <YEAR>, <OWNER>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
1.265 jetty-http 9.4.31.v20200723

1.265.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution” means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor” means any person or entity that distributes the Program.
"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:
a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:
   a) promptly notify the Commercial Contributor in writing of such claim, and
   b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at
its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted
under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient’s rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

================================================================================
Jetty Web Container
The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN
module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

-----

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api
The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.266 log4j-slf4j-impl 2.17.0

1.266.1 Available under license:

Apache Log4j SLF4J Binding
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or
You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.267 httpclient5 5.0
Apache HttpClient
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.268 libnl3 3.2.29 0ubuntu3

1.268.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.
When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".
A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may
distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
1.269 zlib 1.2.11
1.269.1 Available under license :

/* zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly        Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

*/

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.270 lcms 2.10

1.270.1 Available under license:

Little CMS
Copyright (c) 1998-2011 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

THIS LICENSE APPLIES ONLY TO iccjpeg.c file

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-2013, Thomas G. Lane, Guido Vollbeding. All Rights Reserved except as specified below.
Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, ltmain.sh). Another support script, install-sh, is copyright by X Consortium but is also freely distributable.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that

"The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."
1.271 jersey-apache-connector 2.30.1

1.271.1 Available under license :

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses
This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content
Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
* by the @authors tag.

Bootstrap v3.3.7
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.9.9
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* License: MIT & GPL (http://www.opensource.org/licenses/mit-license.php &
http://www.gnu.org/licenses/gpl.html)
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public
  domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.
"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition
of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in
writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the
minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Open Source Used In IOx-FogDirector-AC18 1.17.0 3053
Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code.
And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

Open Source Used In IOx-FogDirector-AC18 1.17.0 3055
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.272 libgpg-error 1.27 6

1.272.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software— to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that
there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.
1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the 
entire whole, and thus to each and every part regardless of who wrote 
it.

Thus, it is not the intent of this section to claim rights or contest 
your rights to work written entirely by you; rather, the intent is to 
exercise the right to control the distribution of derivative or 
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library 
with the Library (or with a work based on the Library) on a volume of 
a storage or distribution medium does not bring the other work under 
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public 
License instead of this License to a given copy of the Library. To do 
this, you must alter all the notices that refer to this License, so 
that they refer to the ordinary GNU General Public License, version 2, 
instead of to this License. (If a newer version than version 2 of the 
ordinary GNU General Public License has appeared, then you can specify 
that version instead if you wish.) Do not make any other change in 
these notices.

Once this change is made in a given copy, it is irreversible for 
that copy, so the ordinary GNU General Public License applies to all 
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of 
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or 
derivative of it, under Section 2) in object code or executable form 
under the terms of Sections 1 and 2 above provided that you accompany 
it with the complete corresponding machine-readable source code, which 
must be distributed under the terms of Sections 1 and 2 above on a 
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy 
from a designated place, then offering equivalent access to copy the 
source code from the same place satisfies the requirement to 
distribute the source code, even though third parties are not 
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the 
Library, but is designed to work with the Library by being compiled or 
linked with it, is called a "work that uses the Library". Such a 
work, in isolation, is not a derivative work of the Library, and 
therefore falls outside the scope of this License.
However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to
apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms
of the ordinary General Public License).

To apply these terms, attach the following notices to the library.
It is safest to attach them to the start of each source file to most
effectively convey the exclusion of warranty; and each file should
have at least the "copyright" line and a pointer to where the full
notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or
your school, if any, to sign a "copyright disclaimer" for the library,
if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James
Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.273 libncurses 6.1-1ubuntu1.18.04
1.273.1 Available under license:
Upstream source https://invisible-island.net/ncurses/ncurses-examples.html

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

Files: *
Licence: X11

Files: aclocal.m4 package
Copyright: 2003-2017,2018 by Thomas E. Dickey
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, distribute with modifications, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-------------------------------------------------------------------------------
Files: install-sh
Copyright: 1994 X Consortium
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

FSF changes to this file are in the public domain.

Calling this script install-sh is preferred over install.sh, to prevent `make' implicit rules from creating a file called install from it when there is no Makefile.
This script is compatible with the BSD install script, but was written from scratch. It can only install one file at a time, a restriction shared with many OS's install programs.

On Debian systems, the complete text of the GNU General Public License can be found in '/usr/share/common-licenses/GPL-2'

-- vile: txtmode file-encoding=utf-8
This is the Debian prepackaged version of the ncurses library and terminfo utilities. ncurses/terminfo was originally written by Pavel Curtis and Zeyd M. Ben-Halim <zmbenhal@netcom.com>, and is currently held by the Free Software Foundation.

This package was put together by Vaidhyanathan G Mayilrangam <vaidhy@debian.org> and Joel Klecker <espy@debian.org>, using sources obtained from ftp://ftp.gnu.org/gnu/ncurses/ncurses-5.0.tar.gz.
Current versions of the ncurses sources are found at ftp://invisible-island.net/ncurses/.

It is based somewhat on work done by Bruce Perens <Bruce@Pixar.com>, David Engel <david@elo.ods.com>. Michael Alan Dorman <mdorman@debian.org>, Richard Braakman <dark@xs4all.nl>, James Troup <jjtroup@comp.brad.ac.uk>, J.H.M. Dassen (Ray) <jdassen@wi.LeidenUniv.nl>, and Galen Hazelwood <galenh@micron.net> over various years.

Copyright (c) 1998-2018 Free Software Foundation, Inc.
Copyright 2001 by Pradeep Padala

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

Copyright (C) 1994 X Consortium

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

Copyright (c) 1980, 1991, 1992, 1993
The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 1996-2018 by Thomas E. Dickey
All Rights Reserved

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE LISTED COPYRIGHT HOLDER(S) BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.
Copyright (c) 1998-2017,2018 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-- vile:txtmode fc=72
-- $Id: COPYING,v 1.6 2018/01/01 12:00:00 tom Exp $
Upstream source https://invisible-island.net/ncurses/ncurses.html
This package is used for testing builds of ncurses.

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

--------------------------------------------------------------------------------
Files: *
Licence: X11

Files: aclocal.m4 package
Copyright: 1996-2017,2018 by Thomas E. Dickey
Licence: X11

Files: doc/html/NCURSES-Programming-HOWTO.html
Copyright: 2001 by Pradeep Padala
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

-------------------------------------------------------------------------------
Files: install-sh
Copyright: 1994 X Consortium
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

-------------------------------------------------------------------------------
Files: progs/tset.c ncurses/tinfo/read_termcap.c
License: BSD

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

-- vile: txtmode file-encoding=utf-8
Upstream source https://invisible-island.net/ncurses/ncurses-examples.html

Current ncurses maintainer: Thomas Dickey <dickey@invisible-island.net>

#########################################################################
Files: *
Licence: X11

Files: aclocal.m4 package
Copyright: 2010-2017,2018 by Thomas E. Dickey
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, distribute with modifications, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

Files: install-sh
Copyright: 1994 X Consortium
Licence: X11

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE X CONSORTIUM BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the X Consortium shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the X Consortium.

FSF changes to this file are in the public domain.

Calling this script install-sh is preferred over install.sh, to prevent `make' implicit rules from creating a file called install from it when there is no Makefile.

This script is compatible with the BSD install script, but was written from scratch. It can only install one file at a time, a restriction shared with many OS's install programs.
On Debian systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2`

-- vile: txtmode file-encoding=utf-8

1.274 jackson-datatype-hibernate 2.12.1

1.274.1 Available under license:

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

1.275 mxparser 1.2.1

1.275.1 Available under license:

Indiana University Extreme! Lab Software License

Version 1.1.1

Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "Indiana University" and "Indiana University Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University" name nor may "Indiana University" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.276 httpcomponents-core 4.4.5

1.276.1 Available under license:
This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls. See http://www.jcip.net
Apache HttpComponents Core
Copyright 2005-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

======================================================================
This project contains annotations in the package org.apache.http.annotation
which are derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls.
See http://www.jcip.net and the Creative Commons Attribution License
(http://creativecommons.org/licenses/by/2.5)
Full text: http://creativecommons.org/licenses/by/2.5/legalcode

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS
PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR
OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS
LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE
BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED
HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its
entirety in unmodified form, along with a number of other contributions, constituting separate and independent
works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be
considered a Derivative Work (as defined below) for the purposes of this License.

"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a
translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art
reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or
adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

"Licensor" means the individual or entity that offers the Work under the terms of this License.

"Original Author" means the individual or entity who created the Work.

"Work" means the copyrightable work of authorship offered under the terms of this License.

"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
- to create and reproduce Derivative Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
- to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:

Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the
terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full
compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

1.277 spring-web 4.3.22.RELEASE

1.277.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2010 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/support/ServletContextResourceLoader.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/http/client/AbstractAsyncClientHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/util/AbstractUriTemplateHandler.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/cors/CorsConfigurationSource.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/context/annotation/RequestScope.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/bind/annotation/ModelAttribute.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/multipart/support/MissingServletRequestPartException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/util/DefaultUriTemplateHandler.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3.22-release-sources-3-jar/org/springframework/web/context/support/ServletContextResourceLoader.java

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/method/annotation/AbstractCookieValueMethodArgumentResolver.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/AsyncClientHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/async/AsyncRequestTimeoutException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/async/DeferredResult.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/support/BasicAuthorizationInterceptor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/OkHttp3ClientHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/annotation/SessionScope.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/support/InterceptingAsyncHttpAccessor.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/remoting/htpinvoke/AbstractHttpInvokerRequestExecutor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/ClientHttpResponse.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/InterceptingClientHttpRequestFactory.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/convert/feed/RssChannelHttpMessageConverter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/OkHttp3ClientHttpResponse.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/filter/CharacterEncodingFilter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/util/UriTemplate.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/support/LiveBeansViewServlet.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/client/AsyncRestTemplate.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/client/UnknownHttpStatusCodeException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/HttpComponentsAsyncClientHttpRequestFactory.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/client/AsyncClientHttpRequestFactory.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2005 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
  jar/org/springframework/web/filter/ServletContextRequestLoggingFilter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
  jar/org/springframework/web/filter/ServletContextRequestLoggingFilter.java

No license file was found, but licenses were detected in source scan.
jar/org/springframework/web/method/annotation/MethodArgumentTypeMismatchException.java

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-
jar/org/springframework/http/client/BufferingClientHttpResponseWrapper.java

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-
jar/org/springframework/web/context/ContextLoaderListener.java

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-
jar/org/springframework/web/context/support/ServletContextResourcePatternResolver.java

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-
jar/org/springframework/web/multipart/MultipartResolver.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-
jar/org/springframework/http/converter/ActivationMediaTypeFactory.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/support/InterceptingHttpAccessor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/multipart/MultipartHttpServletRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/server/ServerHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/AsyncClientHttpRequestFactory.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/ClientHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/method/support/HandlerMethodReturnValueHandler.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/remoting/caucho/SimpleHessianServiceExporter.java

No license file was found, but licenses were detected in source scan.

/
* Copyright 2002-2014 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/method/annotation/SessionStatusMethodArgumentResolver.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/support/ServletContextPropertySource.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/request/async/WebAsyncTask.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/jsf/FacesContextUtils.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/remoting/jaxws/AbstractJaxWsServiceExporter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/AsyncClientHttpRequest.java
/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 */

No license file was found, but licenses were detected in source scan.
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
  jar/org/springframework/http/converter/GenericHttpMessageConverter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2019 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
  jar/org/springframework/http/converter/FormHttpMessageConverter.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2015 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*     http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
No license file was found, but licenses were detected in source scan.

*/

/*
 * Copyright 2002-2009 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/multipart/commons/CommonsMultipartResolver.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/bind/annotation/PathVariable.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/converter/AbstractHttpMessageConverter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/converter/json/AbstractJackson2HttpMessageConverter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http-client/ClientHttpRequestInterceptor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/HttpEntity.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/server/ServletServerHttpRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/HttpHeaders.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/method/support/InvocableHandlerMethod.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/support/GroovyWebApplicationContext.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/converter/xml/AbstractJaxb2HttpMessageConverter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/client/ResponseErrorHandler.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/context/support/WebAsyncManager.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/converter/xml/Jaxb2RootElementHttpMessageConverter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/http/multipart/support/StandardServletMultipartResolver.java

Open Source Used In IOx-FogDirector-AC18 1.17.0 3112
jar/org/springframework/http/HttpStatus.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/remoting/httpinvoker/SimpleHttpInvokerRequestExecutor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/bind/annotation/support/HandlerMethodInvoker.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/client/MessageBodyClientHttpResponseWrapper.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/bind/annotation/RequestMethodArgumentResolver.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/annotation/MapMethodProcessor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/multipart/support/ByteArrayMultipartFileEditor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/filter/Log4jNestedDiagnosticContextFilter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/support/ContextExposingHttpServletRequest.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2008 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/multipart/support/ByteArrayMultipartFileEditor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/filter/Log4jNestedDiagnosticContextFilter.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/annotation/MapMethodProcessor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-jar/org/springframework/web/context/support/ContextExposingHttpServletRequest.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2012 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 *
Open Source Used In IOx-FogDirector-AC18 1.17.0  3117

jar/org/springframework/web/bind/annotation/ValueConstants.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/multipart/MultipartRequest.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/ContextCleanupListener.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/bind/ServletRequestBindingException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/support/ConfigurableWebBindingInitializer.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/client/ResourceAccessException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/support/RequestHandledEvent.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/jsf/DelegatingPhaseListenerMulticaster.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/util/NestedServletException.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/jsf/DecoratingNavigationHandler.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/support/Log4jNestedDiagnosticContextInterceptor.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/http/client/AbstractClientHttpResponse.java
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4-3-22-release-sources-3-
jar/org/springframework/web/context/support/SpringBeanAutowiringSupport.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2007 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141297363_1614968782.46/0/spring-web-4.3-22-release-sources-3-jar/org/springframework/web/bind/support/SessionStatus.java
/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
*/
*/
*/
*/
*/
*/
1.278 rtmpdump 2.4+20151223.gitfa8646d.1 1

1.278.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with
We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
courage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating
system.

Although the Lesser General Public License is Less protective of the
users' freedom, it does ensure that the user of a program that is
linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and
modification follow. Pay close attention to the difference between a
"work based on the library" and a "work that uses the library". The
former contains code derived from the library, whereas the latter must
be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other
program which contains a notice placed by the copyright holder or
other authorized party saying it may be distributed under the terms of
this Lesser General Public License (also called "this License").
Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data
prepared so as to be conveniently linked with application programs
(which use some of those functions and data) to form executables.
The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.
This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties with
this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent
license would not permit royalty-free redistribution of the Library by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program,whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not
If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
This work was packaged for Debian by:

Reinhard Tartler <siretart@tauware.de> on Sun, 30 May 2010 17:07:16 +0200

It was downloaded from http://rtmpdump.mplayerhq.hu/
Upstream Authors and Copyright:

RTMP Dump
(C) 2009 Andrej Stepanchuk
(C) 2009-2011 Howard Chu
(C) 2010 2a665470ced7adb7156fcef47f8199a6371c117b8a79e399a2771e0b36384090

License of the programs in the rtmpdump package:

rtmpdump - small dumper for media content streamed over the RTMP protocol
Copyright (C) 2009 Andrej Stepanchuk
Copyright (C) 2009-2010 Howard Chu

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

License of the librtmp library (included in the librtmp-dev package and in the
librtmp/ subdirectory in the source tree):

Copyright (C) 2005-2008 Team XBMC
http://www.xbmc.org
Copyright (C) 2008-2009 Andrej Stepanchuk
Copyright (C) 2009-2010 Howard Chu

librtmp is free software; you can redistribute it and/or modify
it under the terms of the GNU Lesser General Public License as
published by the Free Software Foundation; either version 2.1,
or (at your option) any later version.

librtmp is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
You should have received a copy of the GNU Lesser General Public License along with librtmp see the file COPYING. If not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.
http://www.gnu.org/copyleft/lgpl.html

The Debian packaging is:

Copyright (C) 2010 Reinhard Tartler <siretart@tauware.de>

and is licensed under the GNU Lesser General Public License.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2' and the text of the GNU Lesser General Public License is in `/usr/share/common-licenses/LGPL-2.1'.

1.279 netty-codec-http 4.1.49.Final
1.279.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*
   * Copyright 2015 The Netty Project
   *
   * The Netty Project licenses this file to you under the Apache License,
   * version 2.0 (the "License"); you may not use this file except in compliance
   * with the License. You may obtain a copy of the License at:
   *
   *   http://www.apache.org/licenses/LICENSE-2.0
   *
   * Unless required by applicable law or agreed to in writing, software
   * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
   * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
   * License for the specific language governing permissions and limitations
   * under the License.
   */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Open Source Used In IOx-FogDirector-AC18 1.17.0

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License, version
 * 2.0 (the "License"); you may not use this file except in compliance with the
 * License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations under
 * the License.
 * /

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-jar/io/netty/handler/codec/http/websockets/WebSocket00FrameDecoder.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

Open Source Used In IOx-FogDirector-AC18 1.17.0 3145
/*
 * Copyright 2017 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/*
 * Adaptation of http://bjoern.hoehrmann.de/utf-8/decoder/dfa/
 *
 * Copyright (c) 2008-2009 Bjoern Hoehrmann <bjoern@hoehrmann.de>
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software
 * and associated documentation files (the "Software"), to deal in the Software without restriction,
 * including without limitation the rights to use, copy, modify, merge, publish, distribute,
 * sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or
 * substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING
 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
 * PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
 * FOR ANY CLAIM,
 * DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
 * ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
 * SOFTWARE.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/http/websocketx/Utf8Validator.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Found in path(s):
/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/multipart/HttpPostRequestDecoder.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpMethod.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/multipart/FileUpload.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/DefaultCookie.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/DefaultHttpResponse.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/spdy/SpdySession.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/rtsp/package-info.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/websocketx/BinaryWebSocketFrame.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpRequest.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpHeaders.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/websocketx/WebSocketClientHandshaker.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/websocketx/WebSocketClientHandshaker08.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/multipart/HttpData.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/websocketx/WebSocket00FrameEncoder.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpRequest.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpHeaders.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/spdy/SpdySession.java

/opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/multipart/HttpPostBodyUtil.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/HttpResponseDecoder.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/multipart/AbstractHttpData.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/TextWebSocketFrame.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/WebSocket07FrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/WebSocketFrame.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/PingWebSocketFrame.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/WebSocket13FrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/LastHttpContent.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

// (BSD License: http://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/WebSocket07FrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io.netty.handler.codec/http/websocketx/WebSocket13FrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
/*
* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpClientUpgradeHandler.java
* /opt/ws_local/PERMITS_SQL/1068497222_1594393092.07/0/netty-codec-http-4-1-49-final-sources-1-
jar/io/netty/handler/codec/http/HttpServerUpgradeHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

// (BSD License: http://www.opensource.org/licenses/bsd-license)
// All rights reserved.
// Redistribution and use in source and binary forms, with or
// * Redistributions of source code must retain the above
// copyright notice, this list of conditions and the
// following disclaimer.
// * Redistributions in binary form must reproduce the above
// following disclaimer in the documentation and/or other
// * Neither the name of the Webbit nor the names of

Found in path(s):

1.280 xz 5.2.2 1.3
1.280.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.
To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the
"Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility
is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.
4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work
under terms of your choice, provided that the terms permit
modification of the work for the customer's own use and reverse
engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the
Library is used in it and that the Library and its use are covered by
this License. You must supply a copy of this License. If the work
during execution displays copyright notices, you must include the
copyright notice for the Library among them, as well as a reference
directing the user to the copy of this License. Also, you must do one
of these things:

a) Accompany the work with the complete corresponding
machine-readable source code for the Library including whatever
changes were used in the work (which must be distributed under
Sections 1 and 2 above); and, if the work is an executable linked
with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the
Library will not necessarily be able to recompile the application
to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system,
rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work
      based on the Library, uncombined with any other library
      facilities. This must be distributed under the terms of the
      Sections above.

   b) Give prominent notice with the combined library of the fact
      that part of it is a work based on the Library, and explaining
      where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Library or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Library (or any work based on the
Library), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the
original licensor to copy, distribute, link with or modify the Library
subject to these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide...
a warranty) and that users may redistribute the program under 
these conditions, and telling the user how to view a copy of this 
License. (Exception: if the Program itself is interactive but 
does not normally print such an announcement, your work based on 
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If 
identifiable sections of that work are not derived from the Program, 
and can be reasonably considered independent and separate works in 
themselves, then this License, and its terms, do not apply to those 
sections when you distribute them as separate works. But when you 
distribute the same sections as part of a whole which is a work based 
on the Program, the distribution of the whole must be on the terms of 
this License, whose permissions for other licensees extend to the 
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest 
your rights to work written entirely by you; rather, the intent is to 
exercise the right to control the distribution of derivative or 
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program 
with the Program (or with a work based on the Program) on a volume of 
a storage or distribution medium does not bring the other work under 
the scope of this License.

3. You may copy and distribute the Program (or a work based on it, 
under Section 2) in object code or executable form under the terms of 
Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable 
      source code, which must be distributed under the terms of Sections 
      1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three 
      years, to give any third party, for a charge no more than your 
      cost of physically performing source distribution, a complete 
      machine-readable copy of the corresponding source code, to be 
      distributed under the terms of Sections 1 and 2 above on a medium 
      customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer 
      to distribute corresponding source code. (This alternative is 
      allowed only for noncommercial distribution and only if you 
      received the program in object code or executable form with such 
      an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the
users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the
library”. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)
b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.
"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an
implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under
the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This
License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and
only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as
part of a transaction in which the right of possession and use of the
User Product is transferred to the recipient in perpetuity or for a
fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied
by the Installation Information. But this requirement does not apply
if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has
been installed in ROM).

The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates
for a work that has been modified or installed by the recipient, or for
the User Product in which it has been modified or installed. Access to a
network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law. If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own
removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of
that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the
terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your
license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the
rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.
If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License,
13. section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

XZ Utils Licensing

Different licenses apply to different files in this package. Here is a rough summary of which licenses apply to which parts of this package (but check the individual files to be sure!):

- liblzma is in the public domain.

- xz, xzdec, and lzmadec command line tools are in the public domain unless GNU getopt_long had to be compiled and linked in from the lib directory. The getopt_long code is under GNU LGPLv2.1+.

- The scripts to grep, diff, and view compressed files have been adapted from gzip. These scripts and their documentation are under GNU GPLv2+.

- All the documentation in the doc directory and most of the XZ Utils specific documentation files in other directories are in the public domain.

- Translated messages are in the public domain.

- The build system contains public domain files, and files that are under GNU GPLv2+ or GNU GPLv3+. None of these files end up in the binaries being built.
- Test files and test code in the tests directory, and debugging utilities in the debug directory are in the public domain.

- The extra directory may contain public domain files, and files that are under various free software licenses.

You can do whatever you want with the files that have been put into the public domain. If you find public domain legally problematic, take the previous sentence as a license grant. If you still find the lack of copyright legally problematic, you have too many lawyers.

As usual, this software is provided "as is", without any warranty.

If you copy significant amounts of public domain code from XZ Utils into your project, acknowledging this somewhere in your software is polite (especially if it is proprietary, non-free software), but naturally it is not legally required. Here is an example of a good notice to put into "about box" or into documentation:

This software includes code from XZ Utils <http://tukaani.org/xz/>.

The following license texts are included in the following files:
- COPYING.LGPLv2.1: GNU Lesser General Public License version 2.1
- COPYING.GPLv2: GNU General Public License version 2
- COPYING.GPLv3: GNU General Public License version 3

Note that the toolchain (compiler, linker etc.) may add some code pieces that are copyrighted. Thus, it is possible that e.g. liblzma binary wouldn't actually be in the public domain in its entirety even though it contains no copyrighted code from the XZ Utils source package.

If you have questions, don't hesitate to ask the author(s) for more information.

**1.281 libiscsi 1.17.0-1.1**

**1.281.1 Available under license:**

Libiscsi components fall under two separate licences

The lib and include directories

```
- The iscsi client library itself, i.e. the lib and include directories, is licenced under GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or
```
(at your option) any later version.

The library also contains one file : lib/md5.c that is under Public Domain.

A copy of LGPL 2.1 is included in the file LICENCE-LGPL-2.1.txt but can also be downloaded from http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

This is the licence that applies to the libiscsi library and its use.

src/ld_iscsi.c
==============
This LD_PRELOAD utility is licenced under GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

The src, examples and test-tool directories EXCEPT src/ld_iscsi.c
==================================================================
The utility and example applications using this library, i.e. the src and the examples directories, are licenced under the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

These are executable applications that link with the libiscsi library.

A copy of GPL 2 is included in the file LICENCE-GPL-2.txt but can also be downloaded from http://www.gnu.org/licenses/old-licenses/gpl-2.0.html

This licence applies only to the source code for the provided applications that link to and use libiscsi.

To avoid any confusion, every source file also contain a licence boilerplate.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free
software—to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation’s software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author’s protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors’ reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections
1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence

Open Source Used In IOx-FogDirector-AC18 1.17.0 3214
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run
that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the
Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
"any later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.282 librokenheimdal 7.5.0+dfsg-1

1.282.1 Available under license:

Copyright (c) 1995 - 2014 Kungliga Tekniska Hgskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Please see info documentation for the complete list of licenses.

Copyright (c) 1992, 1993
The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by
Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: heimdal
Upstream-Contact: heimdal@h5l.org
Source: https://github.com/heimdal/heimdal
Files-Excluded: doc/doxyout
   lib/wind/rfc3490.txt
   lib/wind/rfc3491.txt
   lib/wind/rfc4013.txt
   lib/wind/rfc4518.txt

Files: *
Copyright: 1996-2017 Kungliga Tekniska Hgskolan
2009, 2010 Apple, Inc.
1991, 2013 Massachusetts Institute of Technology
1995-1997 Eric Young <eay@mincom.oz.au>
1988, 1990, 1993 The Regents of the University of California
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Files: lib/wind/rfc3454.txt
Copyright: None
License: none
Not copyrightable; stripped to contain only tables.

Files: lib/wind/rfc3492.txt
Copyright: 2003 The Internet Society
License: custom
Regarding this entire document or any portion of it (including the pseudocode and C code), the author makes no guarantees and is not responsible for any damage resulting from its use. The author grants irrevocable permission to anyone to use, modify, and distribute it in any way that does not diminish the rights of anyone else to use, modify, and distribute it, provided that redistributed derivative works do not contain misleading author or version information. Derivative works need not be licensed under similar terms.

Files: debian/*
Copyright: 1999-2017 Brian May <bam@debian.org>
2017 Dominik George <nik@naturalnet.de>
License: GPL-2+
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text can be found in /usr/share/common-licenses/GPL-2.

#Files: ?
#Copyright: 1992 Simmule Turner and Rich Salz
#License: Spencer-94
# This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.
# .
# Permission is granted to anyone to use this software for any purpose on any computer system, and to alter it and redistribute it freely, subject to the following restrictions:
# .
# 1. The authors are not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.
# .
# 2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources, credits must appear in the documentation.
# .
# 3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.
# .
# 4. This notice may not be removed or altered.
Copyright (c) 1997-2011 Kungliga Tekniska Hgskolan
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (c) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name of the Institute nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
The parts of the libtelnet that handle Kerberos.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading The Regents of the University of California.

libedit

@verbatim

Copyright (c) 1992, 1993 The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Christos Zoulas of Cornell University.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

END

@end verbatim
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading TomsFastMath / LibTomMath

Tom's fast math (bignum support) and LibTomMath

@verbatim
LibTomMath is hereby released into the Public Domain.
@end verbatim
@copynext

@heading Doug Rabson

GSS-API mechglue layer.

@verbatim
Copyright (c) 2005 Doug Rabson
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@verbatim
@copynext

@heading PADL Software Pty Ltd

@table @asis
@item GSS-API CFX, SPNEGO, naming extensions, API extensions.
@item KCM credential cache.
@item HDB LDAP backend.
@end table

@verbatim

Copyright (c) 2003-2011, PADL Software Pty Ltd.
Copyright (c) 2004, Andrew Bartlett.
Copyright (c) 2003 - 2008, Kungliga Tekniska Hgskolan
Copyright (c) 2015, Timothy Pearson.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
Such damage.

@end verbatim
@copynext

@heading Marko Kreen

Fortuna in libhcrypto

@end verbatim
@copynext

@heading NTT (Nippon Telegraph and Telephone Corporation)

Camellia in libhcrypto

@end verbatim
@copynext

Copyright (c) 2005 Marko Kreen
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

@end verbatim
@copynext

Copyright (c) 2006,2007
NTT (Nippon Telegraph and Telephone Corporation) . All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading The NetBSD Foundation, Inc.

vis.c in libroken

@verbatim
Copyright (c) 1999, 2005 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Vincent Rijmen, Antoon Bosselaers, Paulo Barreto

AES in libhcrypto

@verbatim
rijndael-alg-fst.c

@version 3.0 (December 2000)

Optimised ANSI C code for the Rijndael cipher (now AES)

@author Vincent Rijmen <vincent.rijmen@esat.kuleuven.ac.be>
@author Antoon Bosselaers <antoon.bosselaers@esat.kuleuven.ac.be>
@author Paulo Barreto <paulo.barreto@terra.com.br>

This code is hereby placed in the public domain.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@copynext

@heading Apple, Inc

kdc/announce.c

@verbatim

Copyright (c) 2008 Apple Inc. All Rights Reserved.
Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

@end verbatim

@copynext

@heading Richard Outerbridge

DES core in libcrypto

@verbatim

D3DES (V5.09) -

A portable, public domain, version of the Data Encryption Standard.

Written with Symantec’s THINK (Lightspeed) C by Richard Outerbridge. Thanks to: Dan Hoey for his excellent Initial and Inverse permutation code; Jim Gillogly & Phil Karn for the DES key schedule code; Dennis Ferguson, Eric Young and Dana How for comparing notes; and Ray Lau, for humouring me on.


@end verbatim

@copynext
@heading Secure Endpoints Inc

Windows support

@verbatim

Copyright (c) 2009-2015, Secure Endpoints Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim

@copynext

@heading Novell, Inc

lib/hcrypto/test_dh.c

@verbatim

Copyright (c) 2007, Novell, Inc.
Author: Matthias Koenig <mkoenig@suse.de>

@end verbatim
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Novell nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

@end verbatim
@end copyrightend

1.283 metrics 4.1.9

1.283.1 Available under license:

No license file was found, but licenses were detected in source scan.

* This source file is subject to the MIT license that is bundled

Found in path(s):

* /opt/cola/permits/1283431843_1646721330.67/0/beberlei-metrics-v2-0-2-0-g5919a1c-tar-gz/beberlei-metrics-2850e92/src/Beberlei/Metrics/Collector/InfluxDB.php
* /opt/cola/permits/1283431843_1646721330.67/0/beberlei-metrics-v2-0-2-0-g5919a1c-tar-gz/beberlei-metrics-2850e92/src/Beberlei/Metrics/Collector/Librato.php
* /opt/cola/permits/1283431843_1646721330.67/0/beberlei-metrics-v2-0-2-0-g5919a1c-tar-gz/beberlei-metrics-2850e92/src/Beberlei/Metrics/Collector/Zabbix.php
1.284 hk2-api 2.6.1

1.284.1 Available under license:

# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
   where such changes and/or additions to the Program originate from

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 3240
and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright
license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to
obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
   i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

   ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

   iii) does not attempt to limit or alter the recipients’ rights in the Source Code under section 3.2; and

   iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (“notices”) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes
the Program in a commercial product offering, such Contributor
("Commercial Contributor") hereby agrees to defend and indemnify every
other Contributor ("Indemnified Contributor") against any losses,
damages and costs (collectively "Losses") arising from claims, lawsuits
and other legal actions brought by a third party against the Indemnified
Contributor to the extent caused by the acts or omissions of such
Commercial Contributor in connection with its distribution of the Program
in a commercial product offering. The obligations in this section do not
apply to any claims or Losses relating to any actual or alleged
intellectual property infringement. In order to qualify, an Indemnified
Contributor must: a) promptly notify the Commercial Contributor in
writing of such claim, and b) allow the Commercial Contributor to control,
and cooperate with the Commercial Contributor in, the defense and any
related settlement negotiations. The Indemnified Contributor may
participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial
product offering, Product X. That Contributor is then a Commercial
Contributor. If that Commercial Contributor then makes performance
claims, or offers warranties related to Product X, those performance
claims and warranties are such Commercial Contributor's responsibility
alone. Under this section, the Commercial Contributor would have to
defend claims against the other Contributors related to those performance
claims and warranties, and if a court requires any other Contributor to
pay any damages as a result, the Commercial Contributor must pay
those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS"
BASE, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR
IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF
TITLE, NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR
PURPOSE. Each Recipient is solely responsible for determining the
appropriateness of using and distributing the Program and assumes all
risks associated with its exercise of rights under this Agreement,
including but not limited to the risks and costs of program errors,
compliance with applicable laws, damage to or loss of data, programs
or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT
PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS
SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST
PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted
under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

### The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is
covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that
system in reliance on consistent application of that system; it is up to
the author/donor to decide if he or she is willing to distribute
software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be
a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may
add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new
versions of the General Public License from time to time. Such new
versions will be similar in spirit to the present version, but may
differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version"; you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by the
Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

# Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.
* Project home: https://projects.eclipse.org/projects/ee4j.glassfish

## Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/glassfish-ha-api
* https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor
* https://github.com/eclipse-ee4j/glassfish-shoal
* https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck
* https://github.com/eclipse-ee4j/glassfish-jsftemplating
* https://github.com/eclipse-ee4j/glassfish-hk2-extra
* https://github.com/eclipse-ee4j/glassfish-hk2
* https://github.com/eclipse-ee4j/glassfish-fighterfish

## Third-party Content

This project leverages the following third party content.

None

## Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software,
please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.285 javax-enterprise-concurrent-api 1.0

1.285.1 Available under license :

Found license 'General Public License 2.0' in '* Copyright (c) 2010-2012 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in '* Copyright (c) 2010-2011 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

1.286 netty-codec-dns 4.1.49.Final

1.286.1 Available under license :

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2019 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
No license file was found, but licenses were detected in source scan.

~ Copyright 2014 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 * *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068720001_1594427042.39/0/netty-codec-dns-4-1-49-final-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
1.287 jackson 2.1.0

1.287.1 Available under license:

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.
“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.


The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.
The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
   b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

Open Source Used In IOx-FogDirector-AC18 1.17.0 3261
c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of
product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.
An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License.
You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of
conveying software, under which you make payment to the third party based on the extent of your activity of
conveying the work, and under which the third party grants, to any of the parties who would receive the covered
work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you
(or copies made from those copies), or (b) primarily for and in connection with specific products or compilations
that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to
infringement that may otherwise be available to you under applicable patent law.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of
this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as
to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a
consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty
for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms
and this License would be to refrain entirely from conveying the Program.
13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with
a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to
convey the resulting work. The terms of this License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through
a network will apply to the combination as such.
14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from
time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address
new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of
the GNU General Public License “or any later version” applies to it, you have the option of following the terms and
conditions either of that numbered version or of any later version published by the Free Software Foundation. If the
Program does not specify a version number of the GNU General Public License, you may choose any version ever
published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be
used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for
the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are
imposed on any author or copyright holder as a result of your choosing to follow a later version.
15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER
PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS
How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.
If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <https://www.gnu.org/licenses/why-not-lgpl.html>.

1.288 jackson-databind 2.12.1
1.288.1 Available under license :

# Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

## Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0 To find the details that apply to this artifact see the accompanying LICENSE file.

## Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.289 apt 1.6.14
1.289.1 Available under license :

Apt is currently developed by APT Development Team <deity@lists.debian.org>.

License: GPLv2+

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

See /usr/share/common-licenses/GPL-2, or
<http://www.gnu.org/copyleft/gpl.txt> for the terms of the latest version
of the GNU General Public License.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

    Yoyodyne, Inc., hereby disclaims all copyright interest in the program
    `Gnomovision' (which makes passes at compilers) written by James Hacker.

    <signature of Ty Coon>, 1 April 1989
    Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
durnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.290 jna 5.5.0

1.290.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic
mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to which You do not owns copyright or proprietary rights shall be the sole property of the Licensor, and You hereby irrevocably convey all of Your copyright and other intellectual property rights in and to such Contribution to the Licensor and grant to the Licensor the right to grant sublicenses and derivatives of such Contributions on the terms stated in this License.

Open Source Used In IOx-FogDirector-AC18 1.17.0 3282
inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.291 generex 1.0.2

1.291.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2014 y.mifrah
 *
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/GenerexIterator.java
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterator.java
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Node.java
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterable.java
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Main.java
*
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 y.mifrah
*/
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):
/opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexUnitTest.java
No license file was found, but licenses were detected in source scan.

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.github.mifmif</groupId>
  <artifactId>generex</artifactId>
  <version>1.0.2</version>
  <name>Generex</name>
  <url>https://github.com/mifmif/Generex/tree/master</url>
  <description>Generex A Java Library for regex to Strings generation</description>
  <packaging>jar</packaging>
  <licenses>
    <license>
      <name>The Apache Software License, Version 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
      <distribution>repo</distribution>
    </license>
  </licenses>
  <developers>
    <developer>
      <id>mifmif</id>
      <name>MIFRAH Youssef</name>
      <email>mifmif.com@gmail.com</email>
    </developer>
    <developer>
      <id>mkolisnyk</id>
      <name>mkolisnyk</name>
      <email>kolesnik.nickolay@gmail.com</email>
    </developer>
  </developers>
</project>
<parent>
<groupId>org.sonatype.oss</groupId>
<artifactId>oss-parent</artifactId>
<version>7</version>
</parent>

<properties>
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
<maven.pmd.excludes>**/generated-sources/**</maven.pmd.excludes>
</properties>

<scm>
<connection>scm:git:git@github.com:mifmif/Generex.git</connection>
<developerConnection>scm:git:git@github.com:mifmif/Generex.git</developerConnection>
<url>https://github.com/mifmif/Generex.git</url>
</scm>

<ciManagement>
<url>https://travis-ci.org/mifmif/Generex/builds</url>
<system>Travis</system>
<notifiers>
<notifier>
<address>mifmif.com@gmail.com</address>
</notifier>
<notifier>
<address>kolesnik.nickolay@gmail.com</address>
</notifier>
</notifiers>
</ciManagement>

<distributionManagement>
<snapshotRepository>
<id>ossrh</id>
<url>https://oss.sonatype.org/content/repositories/snapshots</url>
</snapshotRepository>
<repository>
<id>ossrh</id>
<url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
</repository>
</distributionManagement>

<build>
<sourceDirectory>src/main/java</sourceDirectory>
<testSourceDirectory>src/test/java</testSourceDirectory>
<resources>
</resources>
<directory>src</directory>
<excludes>
<exclude>*/*.java</exclude>
</excludes>
</resource>

<resource>
<directory>target/dependency</directory>
<excludes>
<exclude>*/*.java</exclude>
</excludes>
</resource>

<resource>
<directory>src/main/resources</directory>
<includes>
<include>*/*.properties</include>
</includes>
</resource>

<plugins>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-install-plugin</artifactId>
<version>2.5.2</version>
<configuration>
<file>target/${project.artifactId}-${project.version}.jar</file>
<groupId>${project.groupId}</groupId>
<artifactId>${project.artifactId}</artifactId>
<version>${project.version}</version>
<packaging>maven-plugin</packaging>
</configuration>
</plugin>

<plugin>
<artifactId>maven-compiler-plugin</artifactId>
<version>3.5.1</version>
<configuration>
<source>1.5</source>
<target>1.5</target>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-jar-plugin</artifactId>
<version>3.0.2</version>
<configuration>
<excludes>
<exclude>*</exclude>
</excludes>
<exclude>com/thoughtworks/**/*</exclude>
</configuration>
</plugin>

<plugin>
<artifactId>maven-compiler-plugin</artifactId>
<version>3.5.1</version>
<configuration>
<source>1.5</source>
<target>1.5</target>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-jar-plugin</artifactId>
<version>3.0.2</version>
<configuration>
<excludes>
<exclude>*</exclude>
<exclude>com/thoughtworks/**/*</exclude>
</configuration>
</plugin>
<exclude>freemarker/**/*</exclude>
<exclude>ftl/**/*</exclude>
<exclude>i18n/**/*</exclude>
<exclude>style/**/*</exclude>
<exclude>junit/**/*</exclude>
<exclude>licenses/**/*</exclude>
<exclude>META-INF/maven/**/*</exclude>
<exclude>org/codehaus/**/*</exclude>
<exclude>org/hamcrest/**/*</exclude>
<exclude>junit/**/*</exclude>
<exclude>org/testng/**/*</exclude>
<exclude>org/xmlpull/**/*</exclude>
<exclude>stories/**/*</exclude>
<exclude>style/**/*</exclude>
<exclude>tests/**/*</exclude>
</excludes>

<archive>
<manifest>
<addClasspath>false</addClasspath>
<addDefaultImplementationEntries>true</addDefaultImplementationEntries>
<addDefaultSpecificationEntries>true</addDefaultSpecificationEntries>
<addExtensions>false</addExtensions>
<classpathLayoutType>simple</classpathLayoutType>
</manifest>
</archive>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-assembly-plugin</artifactId>
<version>2.6</version>
<configuration>
<descriptors>
<descriptor>./sources.xml</descriptor>
</descriptors>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-dependency-plugin</artifactId>
<version>2.10</version>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
<version>2.10.3</version>
<configuration>
<show>private</show>
</configuration>
</plugin>
<nohelp>false</nohelp>
<failOnError>false</failOnError>
<sourcepath>src/main/java</sourcepath>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-release-plugin</artifactId>
<configuration>
<tagNameFormat>v@{project.version}</tagNameFormat>
<preparationGoals>package install:install-file</preparationGoals>
<scmCommentPrefix>#3</scmCommentPrefix>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-deploy-plugin</artifactId>
<version>2.8.2</version>
<configuration>
<skip>true</skip>
</configuration>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
<version>3.0.1</version>
<executions>
<execution>
<goals>
<goal>jar-no-fork</goal>
</goals>
</execution>
</executions>
</plugin>

<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-javadoc-plugin</artifactId>
<executions>
<execution>
<id>attach-javadocs</id>
<goals>
<goal>jar</goal>
</goals>
</execution>
</executions>
</plugin>
<excludes>
<exclude>*/HelpMojo.java</exclude>
</excludes>
</configuration>
</executions>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-checkstyle-plugin</artifactId>
<version>2.10</version>
<configuration>
<excludes>**/test/java/**/*_java</excludes>
<configLocation>./sun_checks.xml</configLocation>
<suppressionsLocation>./checkstyle-suppressions.xml</suppressionsLocation>
<failOnViolation>true</failOnViolation>
</configuration>
</executions>
</plugin>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>cobertura-maven-plugin</artifactId>
<version>2.6</version>
<configuration>
<check>
<haltOnFailure>false</haltOnFailure>
<totalBranchRate>80</totalBranchRate>
<totalLineRate>80</totalLineRate>
</check>
</configuration>
</plugin>
</check>
</instrumentation>
</excludes>
</configuration>
</executions>
</plugin>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>javancss-maven-plugin</artifactId>
<version>2.0</version>
<configuration>
<encoding>${project.build.sourceEncoding}</encoding>
<ccnLimit>10</ccnLimit>
<ncssLimit>100</ncssLimit>
<failOnViolation>true</failOnViolation>
<excludes>
<exclude>**/tests/**/*.*</exclude>
</excludes>
</configuration>
</executions>
</plugin>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>findbugs-maven-plugin</artifactId>
<version>3.0.3</version>
<configuration>
<encoding>${project.build.sourceEncoding}</encoding>
<failOnError>true</failOnError>
<excludeFilterFile>findBugsExclude.xml</excludeFilterFile>
</configuration>
<executions>
<execution>
<goals>
<goal>check</goal>
</goals>
</execution>
</executions>

<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>jdepend-maven-plugin</artifactId>
.VERSION>2.0</version>
</plugin>
</plugins>
</build>
<dependencies>
<dependency>
<groupId>dk.brics.automaton</groupId>
<artifactId>automaton</artifactId>
<version>1.11-8</version>
</dependency>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>4.12</version>
<scope>test</scope>
</dependency>
<dependency>
<groupId>org.hamcrest</groupId>
<artifactId>hamcrest-junit</artifactId>
<version>2.0.0.0</version>
<scope>test</scope>
</dependency>
</dependencies>

Found in path(s):
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/pom.xml

No license file was found, but licenses were detected in source scan.

Generex is licensed under the Apache License, Version 2.0.
http://www.apache.org/licenses/LICENSE-2.0

Found in path(s):
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/README.md
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2015 y.mifrah
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1257332954_1642801959.04/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexIteratorUnitTest.java

1.292 junixsocket-common 2.3.2
1.292.1 Available under license:
No license file was found, but licenses were detected in source scan.

/**
 * junixsocket
 *
 * Copyright 2009-2020 Christian Kohlschütter
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1258507885_1643078430.71/0/junixsocket-common-2-3-2-sources-
1.293 objenesis 1.0

1.293.1 Available under license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
1.294 docker-java-transport 3.2.5
1.294.1 Available under license:
   Apache-2.0

1.295 acl 2.2.52 3build1
1.295.1 Available under license:
   Most components of the "acl" package are licensed under
   Version 2.1 of the GNU Lesser General Public License (see COPYING.LGPL).

   Some components (as annotated in the source) are licensed
   under Version 2 of the GNU General Public License (see below).

   -----------------------------------------------------------------------------

   GNU GENERAL PUBLIC LICENSE
   Version 2, June 1991

   Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
   51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
   Everyone is permitted to copy and distribute verbatim copies
   of this license document, but changing it is not allowed.

   Preamble

   The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

   When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.
To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
This package was debianized by Nathan Scott nathans@debian.org on
Tue, 26 Feb 2002 13:25:26 +1100

It can be downloaded from ftp://acl.bestbits.at/

Copyright:

Copyright (C) 2001 Andreas Gruenbacher.
Copyright (C) 2001-2002 Silicon Graphics, Inc. All Rights Reserved.

You are free to distribute this software under Version 2.1
of the GNU Lesser General Public License.
On Debian systems, refer to /usr/share/common-licenses/LGPL-2.1
for the complete text of the GNU Lesser General Public License.

Certain components (as annotated in the source) are licensed
under the terms of the GNU General Public License.
On Debian systems, the complete text of the GNU General Public
License can be found in /usr/share/common-licenses/GPL file.
Most components of the "acl" package are licensed under
Version 2.1 of the GNU Lesser General Public License (see below).
below.

Some components (as annotated in the source) are licensed
GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them.
with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author’s reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:
a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.296 qemu 2.11+dfsg-1ubuntu7.39
1.296.1 Available under license :

/*
 * FreeBSD strace list
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, see <http://www.gnu.org/licenses/>.
 */
{ TARGET_FREEBSD_NR___acl_check_fd, "__acl_check_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_check_file, "__acl_check_file", "%s(\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_check_link, "__acl_check_link", "%s(\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_fd, "__acl_delete_fd", "%s(%d, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_file, "__acl_delete_file", "%s(\"%s\", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_delete_link, "__acl_delete_link", "%s(\"%s\", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_fd, "__acl_get_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_file, "__acl_get_file", "%s(\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_get_link, "__acl_get_link", "%s(\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_fd, "__acl_set_fd", "%s(%d, %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_file, "__acl_set_file", "%s("%s", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___acl_set_link, "__acl_set_link", "%s(\"%s\", %d, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___semctl, "__semctl", "$c", NULL, NULL },
{ TARGET_FREEBSD_NR___syscall, "__syscall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___sysct1, "__sysct1", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___umtx_op, "__umtx_op", "%s(%d, %d, %d, %#x, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___accept, "__accept", "%s(%d, %#x, %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___access, "__access", "%s(\"%s\", %#o)", NULL, NULL },
{ TARGET_FREEBSD_NR___acct, "__acct", "$c", NULL, NULL },
{ TARGET_FREEBSD_NR___adftime, "__adftime", "\"s\", %d, %d, %#x", NULL, NULL },
{ TARGET_FREEBSD_NR___bind, "__bind", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___break, "__break", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chdir, "__chdir", "%s(\"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___chflags, "__chflags", "\"s\", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chmod, "__chmod", "%s(\"%s\", %#o)", NULL, NULL },
{ TARGET_FREEBSD_NR___chown, "__chown", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___chroot, "__chroot", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_getres, "__clock_getres", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_gettime, "__clock_gettime", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___clock_settime, "__clock_settime", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___close, "__close", "%s(%d)", NULL, NULL },
{ TARGET_FREEBSD_NR___connect, "__connect", "%s(%d, %#x, %d)", NULL, NULL },
{ TARGET_FREEBSD_NR___dup, "__dup", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___dup2, "__dup2", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___eaccess, "__eaccess", "%s(\"%s\", %#x)", NULL, NULL },
{ TARGET_FREEBSD_NR___execve, "__execve", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR___exit, "__exit", "%s(%d)\n", NULL, NULL },
{ TARGET_FREEBSD_NR___extattrctl, "__extattrctl", "%s(\"%s\", %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_delete_fd, "__extattr_delete_fd", "%s(%d, %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_delete_file, "__extattr_delete_file", "%s(\"%s\", %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_delete_link, "__extattr_delete_link", "%s(\"%s\", %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_get_fd, "__extattr_get_fd", "%s(%d, %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_get_file, "__extattr_get_file", "%s(\"%s\", %d, \"%s\")", NULL, NULL },
{ TARGET_FREEBSD_NR___extattr_get_link, "__extattr_get_link", "%s(\"%s\", %d, \"%s\")", NULL, NULL },

{ TARGET_FREEBSD_NR_extattr_list_fd, "extattr_list_fd", "%s(\%d, %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_extattr_list_file, "extattr_list_file", "%s("\%s", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_extattr_list_link, "extattr_list_link", "%s("\%s", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_extattr_set_fd, "extattr_set_fd", "%s(\%d, %d, \"%s\", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_extattr_set_file, "extattr_set_file", "%s("\%s", %d, \"%s\", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_extattr_set_link, "extattr_set_link", "%s("\%s", %d, \"%s\", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_fchdir, "fchdir", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fchflags, "fchflags", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fchmod, "fchmod", "%s(%d,\%o)\", NULL, NULL },
{ TARGET_FREEBSD_NR_fchown, "fchown", "%s(%d,\%d, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_fcntl, "fcntl", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fexecve, "fexecve", NULL, print_execve, NULL },
{ TARGET_FREEBSD_NR_fhopen, "fhopen", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fhstat, "fhstat", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fhstatfs, "fhstatfs", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fence, "flock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_fhstatat, "fhstatat", "%s(%d,\"%s\", %d, %d\%x, %d)\", NULL, NULL },
{ TARGET_FREEBSD_NR_fhstatfs, "fhstatfs", "%s(%d,\%o)\", NULL, NULL },
{ TARGET_FREEBSD_NR_fsync, "fsync", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ftruncate, "ftruncate", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_futimes, "futimes", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getcontext, "getcontext", "getcontext", "%s(%\%x)\", NULL, NULL },
{ TARGET_FREEBSD_NR_getcontext, "getcontext", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_freebsd6_mmap, "freebsd6_mmap", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getegid, "getegid", "%s()\", NULL, NULL },
{ TARGET_FREEBSD_NR_geteuid, "geteuid", "%s()\", NULL, NULL },
{ TARGET_FREEBSD_NR_getfh, "getfh", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getfsstat, "getfsstat", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getgid, "getgid", "%s()\", NULL, NULL },
{ TARGET_FREEBSD_NR_getgroups, "getgroups", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getitimer, "getitimer", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getlogin, "getlogin", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getpeername, "getpeername", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getppid, "getppid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getpriority, "getpriority", "%s(%\%x)\", NULL, NULL },
{ TARGET_FREEBSD_NR_getresgid, "getresgid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getresuid, "getresuid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getrusage, "getrusage", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getsid, "getsid", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getsockopt, "getsockopt", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_gettimeofday, "gettimeofday", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_getuid, "getuid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_ioctl, "ioctl", NULL, print_ioctl, NULL },
{ TARGET_FREEBSD_NR_isetugid, "isetugid", "%s()", NULL, NULL },
{ TARGET_FREEBSD_NR_kevent, "kevent", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_kill, "kill", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_kqueue, "kqueue", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ktrace, "ktrace", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_lchown, "lchown", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_link, "link", "%s(""%s"", "%s")", NULL, NULL },
{ TARGET_FREEBSD_NR_listen, "listen", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_lpathconf, "lpathconf", "%s(""%s", %d)", NULL, NULL },
{ TARGET_FREEBSD_NR_lseek, "lseek", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_lstat, "lstat", "%s(""%s",%p)", NULL, NULL },
{ TARGET_FREEBSD_NR_madvise, "madvise", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mkdir, "mkdir", "%s(""%s",%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR_mkfifo, "mkfifo", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mknod, "mknod", "%s(""%s",%#o,%#x)", NULL, NULL },
{ TARGET_FREEBSD_NR_mlock, "mlock", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mlockall, "mlockall", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_mmap, "mmap", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_nanosleep, "nanosleep", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_nfssvc, "nfssvc", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_open, "open", "%s(""%s",%#x,%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR_openat, "openat", "%s(%d, "%s",%#x,%#o)", NULL, NULL },
{ TARGET_FREEBSD_NR_pathconf, "pathconf", "%s(""%s",%#d)", NULL, NULL },
{ TARGET_FREEBSD_NR_pipe, "pipe", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_poll, "poll", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pread, "pread", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_preadv, "preadv", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_profil, "profil", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_ptrace, "ptrace", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwrite, "pwrite", NULL, NULL, NULL },
{ TARGET_FREEBSD_NR_pwritev, "pwritev", NULL, NULL, NULL },
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TARGET_FREEBSD_NR_quotactl</td>
<td>&quot;quotactl&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_read</td>
<td>&quot;read&quot;, &quot;%s(%d,%#x,%d)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_readlink</td>
<td>&quot;readlink&quot;, &quot;%s(&quot;%s&quot;,%p,%d)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_readv</td>
<td>&quot;readv&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_reboot</td>
<td>&quot;reboot&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_recvfrom</td>
<td>&quot;recvfrom&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_recvmsg</td>
<td>&quot;recvmsg&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_rename</td>
<td>&quot;rename&quot;, &quot;%s(&quot;%s&quot;,%s&quot;)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_revoke</td>
<td>&quot;revoke&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_rfork</td>
<td>&quot;rfork&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_rmdir</td>
<td>&quot;rmdir&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_rtprio_thread</td>
<td>&quot;rtprio_thread&quot;, &quot;%s(%d, %d, %p)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sbrk</td>
<td>&quot;sbrk&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sched_yield</td>
<td>&quot;sched_yield&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_semget</td>
<td>&quot;semget&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_semop</td>
<td>&quot;semop&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sendmsg</td>
<td>&quot;sendmsg&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sendto</td>
<td>&quot;sendto&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setcontext</td>
<td>&quot;setcontext&quot;, &quot;%s(%#x)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setegid</td>
<td>&quot;setegid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setgid</td>
<td>&quot;setgid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setgroups</td>
<td>&quot;setgroups&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setitimer</td>
<td>&quot;setitimer&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setlogin</td>
<td>&quot;setlogin&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setpgid</td>
<td>&quot;setpgid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setpriority</td>
<td>&quot;setpriority&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setregid</td>
<td>&quot;setregid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setresgid</td>
<td>&quot;setresgid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setresuid</td>
<td>&quot;setresuid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setreuid</td>
<td>&quot;setreuid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setrlimit</td>
<td>&quot;setrlimit&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setsid</td>
<td>&quot;setsid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setsockopt</td>
<td>&quot;setsockopt&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_settimeofday</td>
<td>&quot;settimeofday&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_setuid</td>
<td>&quot;setuid&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_shmat</td>
<td>&quot;shmat&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_shmctl</td>
<td>&quot;shmctl&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_shmdt</td>
<td>&quot;shmdt&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_shmget</td>
<td>&quot;shmget&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_shutdown</td>
<td>&quot;shutdown&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigaction</td>
<td>&quot;sigaction&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_signal</td>
<td>&quot;signal&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigsuspend</td>
<td>&quot;sigsuspend&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigreturn</td>
<td>&quot;sigreturn&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigpending</td>
<td>&quot;sigpending&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigprocmask</td>
<td>&quot;sigprocmask&quot;, NULL, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_sigaltstack</td>
<td>&quot;sigaltstack&quot;, &quot;%s(%p,%p)&quot;, NULL, NULL },</td>
</tr>
<tr>
<td>TARGET_FREEBSD_NR_gettime</td>
<td>&quot;gettimeofday&quot;, NULL, NULL, NULL },</td>
</tr>
</tbody>
</table>
NOTE! This copyright does *not* cover user programs that use kernel services by normal system calls - this is merely considered normal use of the kernel, and does *not* fall under the heading of "derived work". Also note that the GPL below is copyrighted by the Free Software Foundation, but the instance of code that it refers to (the Linux kernel) is copyrighted by me and others who actually wrote it.

Also note that the only valid version of the GPL as far as the kernel is concerned is _this_ particular version of the license (ie v2, not v2.2 or v3.x or whatever), unless explicitly otherwise stated.

Linus Torvalds

---------------------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>

Open Source Used In IOx-FogDirector-AC18 1.17.0 3325
Slirp was written by Danny Gasparovski.  
Copyright (c), 1995,1996 All Rights Reserved.

Slirp is maintained by Kelly Price <tygris+slirp@erols.com>

Slirp is free software; "free" as in you don't have to pay for it, and you are free to do whatever you want with it. I do not accept any donations, monetary or otherwise, for Slirp. Instead, I would ask you to pass this potential donation to your favorite charity. In fact, I encourage *everyone* who finds Slirp useful to make a small donation to their favorite charity (for example, GreenPeace). This is not a requirement, but a suggestion from someone who highly values the service they provide.

The copyright terms and conditions:

---BEGIN---

Copyright (c) 1995,1996 Danny Gasparovski. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DANNY GASPAROVSKI OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---END---

This basically means you can do anything you want with the software, except 1) call it your own, and 2) claim warranty on it. There is no warranty for this software. None. Nada. If you lose a million dollars while using Slirp, that's your loss not mine. So, ***USE AT YOUR OWN RISK!***.

If these conditions cannot be met due to legal restrictions (E.g. where it is against the law to give out Software without warranty), you must cease
using the software and delete all copies you have.

Slirp uses code that is copyrighted by the following people/organizations:

Juha Pirkola.
Gregory M. Christy.
The Regents of the University of California.
Carnegie Mellon University.
The Australian National University.
RSA Data Security, Inc.

Please read the top of each source file for the details on the various copyrights.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.
8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Copyright (c) Individual contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PyCA Cryptography nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following points clarify the QEMU license:

1) QEMU as a whole is released under the GNU General Public License, version 2.
2) Parts of QEMU have specific licenses which are compatible with the GNU General Public License, version 2. Hence each source file contains its own licensing information. Source files with no licensing information are released under the GNU General Public License, version 2 or (at your option) any later version.

As of July 2013, contributions under version 2 of the GNU General Public License (and no later version) are only accepted for the following files or directories: bsd-user/, linux-user/, hw/vfio/, hw/xen/xen_pt*.

3) The Tiny Code Generator (TCG) is released under the BSD license (see license headers in files).

4) QEMU is a trademark of Fabrice Bellard.

Fabrice Bellard and the QEMU team

A. HISTORY OF THE SOFTWARE

Python was created in the early 1990s by Guido van Rossum at Stichting Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands as a successor of a language called ABC. Guido remains Python's principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for National Research Initiatives (CNRI, see http://www.cnri.reston.va.us) in Reston, Virginia where he released several versions of the software.

In May 2000, Guido and the Python core development team moved to BeOpen.com to form the BeOpen PythonLabs team. In October of the same year, the PythonLabs team moved to Digital Creations (now Zope Corporation, see http://www.zope.com). In 2001, the Python Software Foundation (PSF, see http://www.python.org/psf/) was formed, a non-profit organization created specifically to own Python-related Intellectual Property. Zope Corporation is a sponsoring member of the PSF.

All Python releases are Open Source (see http://www.opensource.org for the Open Source Definition). Historically, most, but not all, Python releases have also been GPL-compatible; the table below summarizes the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived from</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible?</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td>1.2</td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
</tr>
<tr>
<td>1.3 thru 1.5.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>Version</td>
<td>Patch Level</td>
<td>Year</td>
<td>License</td>
<td>Distribution</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>------</td>
<td>---------</td>
<td>--------------</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2</td>
<td>2.1.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.1</td>
<td>2.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.2</td>
<td>2.2.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.3</td>
<td>2.2.2</td>
<td>2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3</td>
<td>2.2.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.1</td>
<td>2.3</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.2</td>
<td>2.3.1</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.3</td>
<td>2.3.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.4</td>
<td>2.3.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.5</td>
<td>2.3.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4</td>
<td>2.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.1</td>
<td>2.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.2</td>
<td>2.4.1</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.3</td>
<td>2.4.2</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5</td>
<td>2.4</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.7</td>
<td>2.6</td>
<td>2010</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON
============================================================================================================

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

---------------------------------------------
1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.
1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.
7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT

CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2
--------------------------------------------------

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.
To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that
uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

LICENCE
======

The software in this repository is covered by the following licence.

// Copyright 2013, ARM Limited
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are met:
//
// * Redistributions of source code must retain the above copyright notice,
//   this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above copyright notice,
//   this list of conditions and the following disclaimer in the documentation
//   and/or other materials provided with the distribution.
// * Neither the name of ARM Limited nor the names of its contributors may be
//   used to endorse or promote products derived from this software without
//   specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS CONTRIBUTORS "AS IS" AND
// ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
// WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
// DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
// FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
// DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
// SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
// CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
The argparse module was contributed to Python as of Python 2.7 and thus was licensed under the Python license. Same license applies to all files in the argparse package project.

For details about the Python License, please see doc/Python-License.txt.

History
-------

Before (and including) argparse 1.1, the argparse package was licensed under Apache License v2.0.

After argparse 1.1, all project files from the argparse project were deleted due to license compatibility issues between Apache License 2.0 and GNU GPL v2.

The project repository then had a clean start with some files taken from Python 2.7.1, so definitely all files are under Python License now. All the files in this directory and subdirectories are released under a BSD like license (see header in each file). No other license is accepted.

1.297 spring-context 4.3.22.RELEASE

1.297.1 Available under license:

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2009 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * 
 *    http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2011 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/annotation/ScopedProxyCreator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/jmx/export/MBeanExportOperations.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/LifecycleProcessor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2016 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 */
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * Abstract the invocation of a cache operation.
 *
 * <p>Does not provide a way to transmit checked exceptions but
 * provide a special exception that should be used to wrap any
 * exception that was thrown by the underlying invocation.
 * Callers are expected to handle this issue type specifically.
 *
 * @author Stephane Nicoll
 * @since 4.1
 */

No license file was found, but licenses were detected in source scan.

/ *
 * Copyright 2002-2008 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/cache/interceptor/CacheOperationInvoker.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/config/MBeanServerBeanDefinitionParser.java
  No license file was found, but licenses were detected in source scan.

/ *
 * Copyright 2002-2008 the original author or authors.
 * *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/config/MBeanServerBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/remoting/RemoteAccessException.java
* Copyright 2002-2014 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
/**
 * @link FactoryBean that obtains a WebSphere \{ @link javax.management.MBeanServer\}
 * reference through WebSphere's proprietary \{ @code AdminServiceFactory\} API,
 * available on WebSphere 5.1 and higher.
 * 
 * Exposes the \{ @code MBeanServer\} for bean references.
 * This FactoryBean is a direct alternative to \{ @link MBeanServerFactoryBean\},
 * which uses standard JMX 1.2 API to access the platform's MBeanServer.
 * 
 * See the javadocs for WebSphere's
 * \a href="http://bit.ly/UzccDr"\{ @code AdminServiceFactory\}\</a>
 * and \a href="http://bit.ly/TRlX2r"\{ @code MBeanFactory\}\</a>.
 * 
 * @author Juergen Hoeller
 * @author Rob Harrop
 * @since 2.0.3
 * @see javax.management.MBeanServer
 * @see MBeanServerFactoryBean
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/jmx/export/assembler/AbstractConfigurableMBeanInfoAssembler.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2005 the original author or authors.
 * 
 * Licensed under the Apache License, Version 2.0 (the "License");
 */
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*  
* http://www.apache.org/licenses/LICENSE-2.0
*  
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/metadata/ManagedOperation.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2013 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *  
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/config/CronTask.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ConfigurationMethod.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/rmi/JndiRmiClientInterceptor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/datetime/joda/DateTimeFormatterFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/i18n/SimpleTimeZoneAwareLocaleContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/jboss/JBossMCAdapter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/Lazy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/ejb/interceptor/SpringBeanAutowiringInterceptor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/bsh/BshScriptUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/concurrent/DefaultManagedTaskExecutor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ImportSelector.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/concurrent/CustomizableThreadFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/scheduling/concurrent/DateTimeFormatterFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ProfileCondition.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/support/TaskUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/i18n/LocaleContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/FilterType.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/Description.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/format/datetime/joda/JodaTimeContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractRefreshableApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/support/ResourceScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractRefreshableApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/support/ResourceScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractRefreshableApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/support/ResourceScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractRefreshableApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/validation/beanvalidation/OptionalValidatorFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/remoting/rmi/RmiRegistryFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/DeferredImportSelector.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ScopedProxyMode.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/i18n/TimeZoneAwareLocaleContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ImportBeanDefinitionRegistrar.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/config/ScheduledTasksBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/remoting/rmi/RmiClientInterceptorUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/SchedulingTaskExecutor.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2018 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/PropertySource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/interceptor/AbstractCacheResolver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/ConfigurationClassEnhancer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/annotation/ProxyCachingConfiguration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/jmx/export/assembler/AbstractReflectiveMBeanInfoAssembler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/annotation/CacheAnnotationParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/weaving/DefaultContextLoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/AnnotationConfigUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/SmartLifecycle.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/support/AbstractMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/interceptor/CacheProxyFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/validation/MessageCodeFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/AnnotationConfigUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/SmartApplicationListener.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/BeanAnnotationHelper.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/support/AbstractMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/event/DefaultEventListenerFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/interceptor/CacheAspectSupport.java
jar/org/springframework/remoting/rmi/RmiServiceExporter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/interceptor/AbstractFallbackCacheOperationSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/annotation/SpringCacheAnnotationParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/support/PropertySourcesPlaceholderConfigurer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/MBeanExporter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ConfigurationClassParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/support/ScriptFactoryPostProcessor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ConfigurationClassPostProcessor.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2016 the original author or authors.
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* http://www.apache.org/licenses/LICENSE-2.0
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/annotation/Cacheable.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/support/LocalValidatorFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/groovy/GroovyScriptFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/config/ScriptBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/interceptor/CacheEvictOperation.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/ParserStrategyUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/support/GenericApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/support/AbstractResourceBasedMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/context/annotation/ImportRegistry.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/datetime/standard/DateTimeContextHolder.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/LoadTimeWeavingConfiguration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/annotation/EnableScheduling.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/config/PropertyOverrideBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/config/ScriptingDefaultsParser.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2002-2017 the original author or authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

/**
* Indicates that a component is only eligible for registration when all
* {@link #value specified conditions} match.
* 
* A <em>condition</em> is any state that can be determined programmatically
* before the bean definition is due to be registered (see
* {@link Condition} for details).
* 
The {@code @Conditional} annotation may be used in any of the following ways:
* 
* as a type-level annotation on any class directly or indirectly annotated with
* {@link Configuration @Configuration}, including
* classes that are as meta-annotation, for the purpose of composing custom stereotype
* annotations
* as a method-level annotation on any
* method</li>
* </ul>
* 
* If a class is marked with the
* all of the methods, annotations, and
* annotations associated with that
* class will be subject to the conditions.
* 
* NOTE: Inheritance of annotations
* is not supported; any conditions from superclasses or from overridden
* methods will not be considered. In order to enforce these semantics,
* itself is not declared as
* Inherited; furthermore, any
* composed annotation that is meta-annotated with
* must not be declared as Inherited.
* 
* @author Phillip Webb
* @author Sam Brannen
* @since 4.0
* @see Condition
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/Conditional.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2002-2010 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/FileSystemXmlApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/StaticMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jndi/JndiLocatorSupport.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/ClassPathXmlApplicationContext.java
No license file was found, but licenses were detected in source scan.

*/
* Copyright 2002-2007 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
  * limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jndi/JndiAccessor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/SpringModelMBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/metadata/ManagedAttribute.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/WeavingTransformer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/support/RemoteInvocationExecutor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/metadata/AbstractJmxAttribute.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/MBeanExporterListener.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2002-2012 the original author or authors.
* 
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
* 
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/assembler/AutodetectCapableMBeanInfoAssembler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/export/MBeanExporterListener.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/event/ContextStartedEvent.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/config/SchedulerBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/config/ContextNamespaceHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/format/Printer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/format/number/AbstractNumberFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/naming/SelfNaming.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/access/ContextBeanFactoryReference.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/SchedulingException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/support/ObjectNameManager.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/jmx/access/InvalidInvocationException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/cache/interceptor/BeanFactoryCacheOperationSourceAdvisor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jndi/JndiLookupFailureException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/naming/IdentityNamingStrategy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/DelegatingThemeSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/validation/SmartValidator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/groovy/GroovyObjectCustomizer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/UnableToRegisterMBeanException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/remoting/support/DefaultRemoteInvocationExecutor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ScopeMetadata.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/validation/SmartValidator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/ui/context/ThemeSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/remoting/support/DefaultRemoteInvocationExecutor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/config/Task.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/scheduling/config/TaskNamespaceHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/ui/context/support/DelegatingThemeSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/groovy/GroovyObjectCustomizer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/framework/remoting/rmi/RmiClientInterceptor.java
Open Source Used In IOx-FogDirector-AC18 1.17.0 3371
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/rmi/RmiInvocationHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/assembler/MBeanInfoAssembler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/TaskScheduler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/support/DelegatingErrorHandlingRunnable.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/BindException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/Errors.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/config/SpringConfiguredBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jndi/JndiTemplate.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/SimpleLoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jndi/JndiTemplateEditor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/support/StaticScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/support/RegistrationPolicy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/Errors.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/support/StaticScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/support/RegistrationPolicy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/SimpleLoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/config/SpringConfiguredBeanDefinitionParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/MBeanServerNotFoundException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/annotation/SchedulingConfigurer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/ui/context/Theme.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/weaving/LoadTimeWeaverAware.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/access/MBeanProxyFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/event/ApplicationContextEvent.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/assembler/SimpleReflectiveMBeanInfoAssembler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/access/DefaultLocatorFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/notification/NotificationListenerBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/interceptor/NameMatchCacheOperationSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/notification/NotificationPublisher.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/interceptor/CacheInterceptor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/Jsr330ScopeMetadataResolver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jndi/JndiObjectTargetSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/ApplicationContextException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/JmxException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/support/DelegatingMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/annotation/AnnotationMBeanExporter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/support/DefaultRemoteInvocationFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/event/ContextRefreshedEvent.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/weaving/LoadTimeWeaverAwareProcessor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jndi/JndiCallback.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/support/DelegatingMessageSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/SchedulingAwareRunnable.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/validation/Validator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/naming/ObjectNamingStrategy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/ui/context/support/UiApplicationContextUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/ApplicationContextAware.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/EnableMBeanExport.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/metadata/ManagedResource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/support/MBeanServerFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scripting/support/RefreshableScriptTargetSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/instrument/classloading/ResourceOverridingShadowingClassLoader.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/remoting/support/RemoteInvocationTraceInterceptor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/scheduling/Trigger.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/ejb/access/AbstractRemoteSlsbInvokerInterceptor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/ApplicationObjectSupport.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/remoting/support/RemoteInvocationUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jmx/export/naming/MetadataNamingStrategy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/expression/BeanExpressionContextAccessor.java
/*
 * Copyright 2002-2017 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License”);
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

No license file was found, but licenses were detected in source scan.

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/Phased.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/ScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/RemoteProxyFailureException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/BindingResultUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/LoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/event/ApplicationListenerMethodAdapter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/number/NumberFormatAnnotationFormatterFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/concurrent/ScheduledExecutorTask.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/datetime/joda/JodaTimeFormatterRegistrar.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ClassPathScanningCandidateComponentProvider.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/support/NotificationListenerHolder.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/websphere/WebSphereClassPreDefinePlugin.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/stereotype/Service.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/annotation/EnableCaching.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/Phased.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/ScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/RemoteProxyFailureException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/BindingResultUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/LoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/event/ApplicationListenerMethodAdapter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/ScriptSource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/RemoteProxyFailureException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/BindingResultUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/instrument/classloading/LoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
jar/org/springframework/cache/interceptor/CacheOperation.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/EnvironmentAware.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/MBeanExportConfiguration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/ConditionContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/validation/DefaultMessageCodesResolver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/Configuration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/stereotype/Controller.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/jndi/TypeMismatchNamingException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/MessageSourceSupport.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/annotation/DateTimeFormat.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/org/springframework/context/support/AbstractApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/jndi/TypeMismatchNamingException.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3.jar/
jar/org/springframework/context/annotation/ContextAnnotationAutowireCandidateResolver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/event/EventExpressionEvaluator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/scheduling/concurrent/ConcurrentTaskScheduler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/scheduling/support/CronSequenceGenerator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/event/GenericApplicationListenerAdapter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/support/ApplicationListenerDetector.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/cache/interceptor/NamedCacheResolver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/jmx/export/naming/KeyNamingStrategy.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/jmx/export/metadata/JmxMetadataUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/support/PostProcessorRegistrationDelegate.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/instrument/classloading/weblogic/WebLogicClassLoaderAdapter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/annotation/ComponentScanAnnotationParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/annotation/Profile.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/scheduling/concurrent/ThreadPoolTaskScheduler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/scheduling/annotation/AsyncAnnotationAdvisor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/cache/config/CacheNamespaceHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/annotation/ComponentScanAnnotationParser.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/context/annotation/Profile.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/format/datetime/standard/DateTimeFormatterFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
ar/jar/org/springframework/scheduling/support/RemoteExporter.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2015 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 */
* http://www.apache.org/licenses/LICENSE-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/format/datetime/standard/MonthDayFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/scripting/bsh/BshScriptEvaluator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/validation/beanvalidation/SpringConstraintValidatorFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/jmx/export/annotation/ManagedMetric.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/context/event/EventExpressionRootObject.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/jmx/access/ConnectorDelegate.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/instrument/classloading/jboss/JBossLoadTimeWeaver.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4.3.22-release-sources-3.jar/org/springframework/format/datetime/joda/PeriodFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/config/LangNamespaceHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/number/NumberFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/support/AbstractCacheManager.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/validation/support/BindingAwareModelMap.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/annotation/ManagedAttribute.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/event/EventListenerFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ImportResource.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/concurrent/ScheduledExecutorFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scripting/support/StandardScriptEvaluator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/format/datetime/joda/YearMonthFormatter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/context/annotation/ProxyAsyncConfiguration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/jmx/export/annotation/ManagedOperationParameters.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/ui/ExtendedModelMap.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/scheduling/annotation/ProxyAsyncConfiguration.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/cache/annotation/CacheConfig.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-jar/org/springframework/remoting/RemoteTimeoutException.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2014 the original author or authors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *      http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 */
* See the License for the specific language governing permissions and
  limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/cache/interceptor/SimpleCacheErrorHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/support/ContextTypeMatchClassLoader.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/cache/interceptor/CacheErrorHandler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/scheduling/concurrent/ForkJoinPoolFactoryBean.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/annotation/ConditionEvaluator.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/validation/ValidationUtils.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/remoting/support/RemoteInvocation.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/support/AbstractRefreshableConfigApplicationContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/annotation/ScannedGenericBeanDefinition.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/datetime/standard/DateTimeContext.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/context/expression/BeanFactoryAccessor.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/jmx/export/assembler/MethodExclusionMBeanInfoAssembler.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/FormatterRegistry.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/scheduling/annotation/AsyncConfigurer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/datetime/standard/TemporalAccessorPrinter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/AnnotationFormatterFactory.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/datetime/standard/TemporalAccessorPrinter.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/format/FormatterRegistry.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/scheduling/annotation/AsyncConfigurer.java
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2002-2006 the original author or authors.
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 * http://www.apache.org/licenses/LICENSE-2.0
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

Found in path(s):
* /opt/cola/permits/1141298164_1614968959.43/0/spring-context-4-3-22-release-sources-3-
  jar/org/springframework/validation/MapBindingResult.java

1.298 spring-expression 5.2.9.RELEASE

1.298.1 Available under license:
Spring Framework 5.2.9.RELEASE
Copyright (c) 2002-2020 Pivotal, Inc.

This product is licensed to you under the Apache License, Version 2.0
(the "License"). You may not use this product except in compliance with
the License.

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for
these subcomponents is subject to the terms and conditions of the
subcomponent's license, as noted in the license.txt file.

Apache License
Version 2.0, January 2004
https://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License” shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

https://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=======================================================================
SPRING FRAMEWORK 5.2.9.RELEASE SUBCOMPONENTS:

Spring Framework 5.2.9.RELEASE includes a number of subcomponents with separate copyright notices and license terms. The product that includes this file does not necessarily use all the open source subcomponents referred to below. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.


Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>

>>> CGLIB 3.3 (cglib:cglib:3.3):

Copyright (c) 1999-2009, OW2 Consortium <https://www.ow2.org/>
Per the LICENSE file in the CGLIB JAR distribution downloaded from https://github.com/cglib/cglib/releases/download/RELEASE_3_3_0/cglib-3.3.0.jar, CGLIB 3.3 is licensed under the Apache License, version 2.0, the text of which is included above.

>>> Objenesis 3.1 (org.objenesis:objenesis:3.1):

Per the LICENSE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html, Objenesis 3.1 is licensed under the Apache License, version 2.0, the text of which is included above.

Per the NOTICE file in the Objenesis ZIP distribution downloaded from http://objenesis.org/download.html and corresponding to section 4d of the Apache License, Version 2.0, in this case for Objenesis:

Objenesis
Copyright 2006-2019 Joe Walnes, Henri Tremblay, Leonardo Mesquita

===============================================================================

To the extent any open source components are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from https://spring.io/projects, Pivotal's website at https://network.pivotal.io/open-source, or by sending a request, with your name and address to: Pivotal Software, Inc., 875 Howard Street, 5th floor, San Francisco, CA 94103, Attention: General Counsel. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. Pivotal can mail a copy of the Source Files to you on a CD or equivalent physical medium.

This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product. Alternatively, the Source Files may accompany the Software.

1.299 byte-buddy 1.10.10
1.299.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION.
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
Copyright ${project.inceptionYear} - ${current.year} ${copyright.holder}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.300 sqlite 3.22.0-1ubuntu0.4

1.300.1 Available under license:
The author disclaims copyright to this source code. In place of a legal notice, here is a blessing:
May you do good and not evil.
May you find forgiveness for yourself and forgive others.
May you share freely, never taking more than you give.

1.301 libpsl 0.19.1-5build1

1.301.1 Available under license:

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: libpsl
Source: https://github.com/rockdaboot/libpsl

Files: *
Copyright: 2014-2016 Tim Ruehsen
License: MIT

Files: debian/*
Copyright: 2014-2016 Daniel Kahn Gillmor
License: MIT

Files: src/psl-make-dafsa src/lookup_string_in_fixed_set.c
Copyright: 2014-2015 The Chromium Authors
License: Chromium

License: MIT
* Permission is hereby granted, free of charge, to any person obtaining a
* copy of this software and associated documentation files (the "Software"),
* to deal in the Software without restriction, including without limitation
* the rights to use, copy, modify, merge, publish, distribute, sublicense,
* and/or sell copies of the Software, and to permit persons to whom the
* Software is furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in
* all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
* FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
* DEALINGS IN THE SOFTWARE.

License: Chromium
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:
*
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.302 spring-hateoas 0.25.1.RELEASE
1.302.1 Available under license :
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=======================================================================

To the extent any open source subcomponents are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by downloading the Source Files from http://www.springsource.org/download, or by sending a request, with your name and address to: VMware, Inc., 3401 Hillview Avenue, Palo Alto, CA 94304, United States of America or email info@vmware.com. All such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General Counsel. VMware shall mail a copy of the Source Files to you on a CD or equivalent
physical medium. This offer to obtain a copy of the Source Files is valid for three years from the date you acquired this Software product.

Spring HATEOAS 0.22
Copyright (c) [2012-2016] Pivotal Software, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License"). You may not use this product except in compliance with the License.

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE file.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.303 aspect-j 1.8.7

1.303.1 Available under license:

Found license 'Eclipse Public License 1.0' in 'All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v1.0'

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular
Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this
Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

**1.304 hostname 3.20**

**1.304.1 Available under license :**

This package was written by Peter Tobias <tobias@et-inf.fho-emden.de>
on Thu, 16 Jan 1997 01:00:34 +0100.

License:

Copyright (C) 2009 Michael Meskes <meskes@debian.org>
Copyright (C) 2004-2005 Graham Wilson <graham@debian.org>
Copyright (C) 1997 Bernd Eckenfels
Copyright (C) 1997 Peter Tobias <tobias@et-inf.fho-emden.de>
Copyright (C) 1996 Free Software Foundation, Inc.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 of the License.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian systems, the complete text of the GNU General Public License can be found in /usr/share/common-licenses/GPL-2 file.

hostname -- set the host name or show the host/domain name

Copyright (C) 1994-1997 Peter Tobias <tobias@et-inf.fho-emden.de>
2009- Michael Meskes <meskes@debian.org>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

1.305 okhttp 4.8.1
1.305.1 Available under license:

Note that publicsuffixes.gz is compiled from The Public Suffix List:
https://publicsuffix.org/list/public_suffix_list.dat

It is subject to the terms of the Mozilla Public License, v. 2.0:
https://mozilla.org/MPL/2.0/

1.306 httpcomponents-core 4.4.13
1.306.1 Available under license:

Apache HttpCore
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, public display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.307 jradius-core 1.1.6-SNAPSHOOT

1.307.1 Available under license:

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or
else, saying that you provide a warranty) and that users may redistribute
the program under these conditions, and telling the user how to view a copy
of this License. (Exception: if the Program itself is interactive but does
not normally print such an announcement, your work based on the Program is
not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable
sections of that work are not derived from the Program, and can be reasonably
considered independent and separate works in themselves, then this License, and
its terms, do not apply to those sections when you distribute them as separate
works. But when you distribute the same sections as part of a whole which is a
work based on the Program, the distribution of the whole must be on the terms
of this License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your
rights to work written entirely by you; rather, the intent is to exercise the
right to control the distribution of derivative or collective works based on
the Program.

In addition, mere aggregation of another work not based on the Program with the
Program (or with a work based on the Program) on a volume of a storage or
distribution medium does not bring the other work under the scope of this
License.

3. You may copy and distribute the Program (or a work based on it, under
Section 2) in object code or executable form under the terms of Sections 1 and
2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source
code, which must be distributed under the terms of Sections 1 and 2 above
on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to
give any third party, for a charge no more than your cost of physically
performing source distribution, a complete machine-readable copy of the
corresponding source code, to be distributed under the terms of Sections 1
and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed only
for noncommercial distribution and only if you received the program in
object code or executable form with such an offer, in accord with
Subsection b above.)

The source code for a work means the preferred form of the work for making
modifications to it. For an executable work, complete source code means all
the source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and
the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE
PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w`. This is free software, and you are welcome to redistribute it under certain conditions; type `show c` for details.

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL, but only where Sun has expressly included in the particular source file's header the words "Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your
Open Source Used In IOx-FogDirector-AC18 1.17.0  3423

choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

==========================================================================

Coova JRadius License:

The Coova extensions to JRadius library are licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within in the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/. See source files for details regarding stand-alone applications contained herein which are released under the GPL.

Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

==========================================================================

Original JRadius License:

The JRadius core library is licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within in the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/

This software is OSI Certified Open Source Software. OSI Certified is a certification mark of the Open Source Initiative.

Stand-alone JRadius Applications:

* JRadiusSimulator
* JRadiusWiFiClient
* RadClient
* RadBench
* RadiusDictionary

Copyright 2004-2006 PicoPoint, B.V.
Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

==========================================================================
Coova JRadius License:

The Coova JRadius core libraries are licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within in the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://www.coova.org/JRadius. See source files for details regarding stand-alone applications contained herein which are released under the GPL.

Copyright 2006-2010 Coova Technologies, LLC. All rights reserved.

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

Original JRadius License:

The JRadius core library is licensed under the The GNU Library or "Lesser" General Public License (LGPL), while stand-alone applications within in the JRadius project, listed below, are released under the standard The GNU General Public License (GPL). For details, visit http://jradius.net/

This software is OSI Certified Open Source Software. OSI Certified is a certification mark of the Open Source Initiative.

Stand-alone JRadius Applications:

* JRadiusSimulator
* JRadiusWiFiClient
* RadClient
* RadBench
* RadiusDictionary

Copyright 2004-2006 PicoPoint, B.V.
Copyright 2006-2008 David Bird <david@coova.com>

See doc/GPL-LICENSE, doc/LGPL-LICENSE, and source files for details.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

(This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.)

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating
Although the Lesser General Public License is less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification").

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.
1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.
However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood
that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

{description}

Copyright (C) {year} {fullname}

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.
You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

{signature of Ty Coon}, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The GNU General Public License (GPL)
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute
For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and
appropriately publish on each copy an appropriate copyright notice and
disclaimer of warranty; keep intact all the notices that refer to this
License and to the absence of any warranty; and give any other recipients
of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you
may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it,
thus forming a work based on the Program, and copy and distribute such
modifications or work under the terms of Section 1 above, provided that
you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating
that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole
or in part contains or is derived from the Program or any part thereof,
to be licensed as a whole at no charge to all third parties under the
terms of this License.

c) If the modified program normally reads commands interactively when
run, you must cause it, when started running for such interactive use in
the most ordinary way, to print or display an announcement including an
appropriate copyright notice and a notice that there is no warranty (or
else, saying that you provide a warranty) and that users may
redistribute the program under these conditions, and telling the user
how to view a copy of this License. (Exception: if the Program itself is
interactive but does not normally print such an announcement, your work
based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable
sections of that work are not derived from the Program, and can be
reasonably considered independent and separate works in themselves, then
this License, and its terms, do not apply to those sections when you
distribute them as separate works. But when you distribute the same sections
as part of a whole which is a work based on the Program, the distribution of
the whole must be on the terms of this License, whose permissions for other
licensees extend to the entire whole, and thus to each and every part
regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your
rights to work written entirely by you; rather, the intent is to exercise
the right to control the distribution of derivative or collective works
based on the Program.

In addition, mere aggregation of another work not based on the Program with
the Program (or with a work based on the Program) on a volume of a storage
or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed
it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an
explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU_ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

GNU Lesser General Public License

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.  59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users’ freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run. TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the
source code for all modules it contains, plus any associated interface
definition files, plus the scripts used to control compilation and
installation of the library.

Activities other than copying, distribution and modification are not covered
by this License; they are outside its scope. The act of running a program
using the Library is not restricted, and output from such a program is
covered only if its contents constitute a work based on the Library
(independent of the use of the Library in a tool for writing it). Whether
that is true depends on what the Library does and what the program that uses
the Library does.

1. You may copy and distribute verbatim copies of the Library's complete
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the notices
that refer to this License and to the absence of any warranty; and
distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you
may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it,
thus forming a work based on the Library, and copy and distribute such
modifications or work under the terms of Section 1 above, provided that you
also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating
that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to
all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table
of data to be supplied by an application program that uses the facility,
other than as an argument passed when the facility is invoked, then you
must make a good faith effort to ensure that, in the event an
application does not supply such function or table, the facility still
operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a
purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must be
optional: if the application does not supply it, the square root
function must still compute square roots.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.
5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if
the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at runtime a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
a) Accompany the combined library with a copy of the same work based on
the Library, uncombined with any other library facilities. This must be
distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part
of it is a work based on the Library, and explaining where to find the
accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the
Library except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense, link with, or distribute the Library
is void, and will automatically terminate your rights under this
License. However, parties who have received copies, or rights, from you
under this License will not have their licenses terminated so long as such
parties remain in full compliance.

9. You are not required to accept this License, since you have not signed
it. However, nothing else grants you permission to modify or distribute the
Library or its derivative works. These actions are prohibited by law if you
do not accept this License. Therefore, by modifying or distributing the
Library (or any work based on the Library), you indicate your acceptance of
this License to do so, and all its terms and conditions for copying,
distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the
Library), the recipient automatically receives a license from the original
licensor to copy, distribute, link with or modify the Library subject to
these terms and conditions. You may not impose any further restrictions on
the recipients' exercise of the rights granted herein. You are not
responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot distribute so
as to satisfy simultaneously your obligations under this License and any
other pertinent obligations, then as a consequence you may not distribute
the Library at all. For example, if a patent license would not permit
royalty-free redistribution of the Library by all those who receive copies
directly or indirectly through you, then the only way you could satisfy both
it and this License would be to refrain entirely from distribution of the
Library.

If any portion of this section is held invalid or unenforceable under any
particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents
or other property right claims or to contest validity of any such claims;
this section has the sole purpose of protecting the integrity of the free
software distribution system which is implemented by public license
practices. Many people have made generous contributions to the wide range of
software distributed through that system in reliance on consistent
application of that system; it is up to the author/donor to decide if he or
she is willing to distribute software through any other system and a
licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a
consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain
countries either by patents or by copyrighted interfaces, the original
copyright holder who places the Library under this License may add an
explicit geographical distribution limitation excluding those countries, so
that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of
the Lesser General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Library does not specify a license version
number, you may choose any version ever published by the Free Software
Foundation.

14. If you wish to incorporate parts of the Library into other free programs
whose distribution conditions are incompatible with these, write to the
author to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals of
preserving the free status of all derivatives of our free software and of
promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR
THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO
THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY
PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR
CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO
LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR
THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER
SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES. END OF TERMS AND CONDITIONS How to Apply These
Terms to Your New Libraries If you develop a new library, and you want it to
be of the greatest possible use to the public, we recommend making it free
software that everyone can redistribute and change. You can do so by
permitting redistribution under these terms (or, alternatively, under the
terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and an idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it
under the terms of the GNU Lesser General Public License as published by
the Free Software Foundation; either version 2.1 of the License, or (at
your option) any later version.

This library is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser
General Public License for more details.

You should have received a copy of the GNU Lesser General Public License
along with this library; if not, write to the Free Software Foundation,
Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:
Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

signature of Ty Coon, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.308 commons-fileupload 1.3.3
1.308.1 Available under license:
Apache Commons FileUpload
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.309 debconf 1.5.66ubuntu1

1.310 keyutils 1.5.9 9.2ubuntu2

1.310.1 Available under license:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.
When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a
copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.
8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THElibrary TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.
We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.
You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

   Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

   In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program ‘Gnomovision’ (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.311 lsb 9.20170808ubuntu1
1.311.1 Available under license :
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: LSB implementation package

Files: *
Copyright: 2002-2010, Chris Lawrence <lawrenc@debian.org>
License: GPL-2

Files: init-functions.d/50-ubuntu-logging
Copyright: 2005-2011, Canonical Ltd.
License: GPL-2

Files: init-functions
Copyright: 2002-2009, Chris Lawrence <lawrenc@debian.org>
License: BSD-3-clause
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the author nor the names of other contributors
may be used to endorse or promote products derived from this software
without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: GPL-2
This program is free software; you can redistribute it
and/or modify it under the terms of the GNU General Public
License as published by the Free Software Foundation;

This program is distributed in the hope that it will be
useful, but WITHOUT ANY WARRANTY; without even the implied
warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE. See the GNU General Public License for more
details.

You should have received a copy of the GNU General Public
License along with this package; if not, write to the Free
Software Foundation, Inc., 51 Franklin St, Fifth Floor,
Boston, MA 02110-1301 USA

On Debian systems, the full text of the GNU General Public
License version 2 can be found in the file
'/usr/share/common-licenses/GPL-2'.

1.312 gzip 1.6 5ubuntu1.1
1.312.1 Available under license:
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited
permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.
5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium
customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it. (Additional permissions may be written to require their own
removal in certain cases when you modify the work.) You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you
add to a covered work, you may (if authorized by the copyright holders of
that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the
terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or
author attributions in that material or in the Appropriate Legal
Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or
requiring that modified versions of such material be marked in
reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or
authors of the material; or

e) Declining to grant rights under trademark law for use of some
trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that
material by anyone who conveys the material (or modified versions of
it) with contractual assumptions of liability to the recipient, for
any liability that these contractual assumptions directly impose on
those licensors and authors.

All other non-permissive additional terms are considered “further
restrictions” within the meaning of section 10. If the Program as you
received it, or any part of it, contains a notice stating that it is
governed by this License along with a term that is a further
restriction, you may remove that term. If a license document contains
a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms
of that license document, provided that the further restriction does
not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you
must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.
Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to
sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.
THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE
USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided
above cannot be given local legal effect according to their terms,
reviewing courts shall apply local law that most closely approximates
an absolute waiver of all civil liability in connection with the
Program, unless a warranty or assumption of liability accompanies a
copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by

Open Source Used In IOx-FogDirector-AC18 1.17.0 3488
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
@c The GNU Free Documentation License.
@center Version 1.3, 3 November 2008

@c This file is intended to be included within another document,
@c hence no sectioning command or @node.

@display
@uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document \textit{free} in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall
directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material
this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties; any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

@item
VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

@item
COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify
you as the publisher of these copies. The front cover must present
the full title with all words of the title equally prominent and
visible. You may add other material on the covers in addition.
Copying with changes limited to the covers, as long as they preserve
the title of the Document and satisfy these conditions, can be treated
as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit
legibly, you should put the first ones listed (as many as fit
reasonably) on the actual cover, and continue the rest onto adjacent
pages.

If you publish or distribute Opaque copies of the Document numbering
more than 100, you must either include a machine-readable Transparent
copy along with each Opaque copy, or state in or with each Opaque copy
a computer-network location from which the general network-using
public has access to download using public-standard network protocols
a complete Transparent copy of the Document, free of added material.
If you use the latter option, you must take reasonably prudent steps,
when you begin distribution of Opaque copies in quantity, to ensure
that this Transparent copy will remain thus accessible at the stated
location until at least one year after the last time you distribute an
Opaque copy (directly or through your agents or retailers) of that
edition to the public.

It is requested, but not required, that you contact the authors of the
Document well before redistributing any large number of copies, to give
them a chance to provide you with an updated version of the Document.

@item MODIFICATIONS

You may copy and distribute a Modified Version of the Document under
the conditions of sections 2 and 3 above, provided that you release
the Modified Version under precisely this License, with the Modified
Version filling the role of the Document, thus licensing distribution
and modification of the Modified Version to whoever possesses a copy
of it. In addition, you must do these things in the Modified Version:

@enumerate A
@item Use in the Title Page (and on the covers, if any) a title distinct
from that of the Document, and from those of previous versions
(which should, if there were any, be listed in the History section
of the Document). You may use the same title as a previous version
if the original publisher of that version gives permission.

@item
List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

@item
State on the Title page the name of the publisher of the Modified Version, as the publisher.

@item
Preserve all the copyright notices of the Document.

@item
Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

@item
Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

@item
Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled ``History'', Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled ``History'' in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the ``History'' section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

@item
For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

@item
Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

@item
Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

@item
Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

@item
Preserve any Warranty Disclaimers.
@end enumerate

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—f. example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.
COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ``History'' in the various original documents, forming one section Entitled ``History''; likewise combine any sections Entitled ``Acknowledgements'', and any sections Entitled ``Dedications''. You must delete all sections Entitled ``Endorsements.``

COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit.
When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.
Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

@item
FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License ``or any later version'' applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

@item
RELICENSING

``Massive Multiauthor Collaboration Site'' (or ``MMC Site'') means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A ``Massive Multiauthor Collaboration'' (or ``MMC'') contained in the site means any set of copyrightable works thus published on the MMC site.

``CC-BY-SA'' means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit
corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

``Incorporate'' means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is ``eligible for relicensing'' if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

@end enumerate

@page
@heading ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

@smallexample
@group
Copyright (C) @var{year}  @var{your name}.
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.  A copy of the license is included in the section entitled ``GNU Free Documentation License''.
@end group
@end smallexample

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the ```with@dots{}Texts.@: line with this:

@smallexample
@group
with the Invariant Sections being @var{list their titles}, with the Front-Cover Texts being @var{list}, and with the Back-Cover Texts being @var{list}.
@end group
@end smallexample
If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

@c Local Variables:
@c ispell-local-pdict: "ispell-dict"
@c End:

### 1.313 resourcelocator 1.0.3
#### 1.313.1 Available under license:
Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'
Found license 'General Public License 2.0' in '* Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

### 1.314 datasource-proxy 1.4.1
#### 1.314.1 Available under license:
No license file was found, but licenses were detected in source scan.

<name>MIT License</name>

Found in path(s):
* /opt/cola/permits/1283431886_1646721486.27/0/datasource-proxy-1-4-1-sources-jar/META-INF/maven/net.ttddyy/datasource-proxy/pom.xml

### 1.315 adduser 3.116ubuntu1
#### 1.315.1 Available under license:
This package was first put together by Ian Murdock <imurdock@debian.org> and was maintained by Steve Phillips <sjp@cvfn.org> from sources written for the Debian Project by Ian Murdock, Ted Hajek <tedhajek@boombox.micro.umn.edu>, and Sven Rudolph <sr1@inf.tu-dresden.de>.

Since Nov 27 1996, it was maintained by Guy Maor <maor@debian.org>. He
Since May 20 2000, it is maintained by Roland Bauerschmidt <rb@debian.org>.

Since March 24 2004, it is maintained by Roland Bauerschmidt <rb@debian.org>, and co-maintained by Marc Haber <mh+debian-packages@zugschlus.de>

Since 23 Oct 2005, it has been maintained by Joerg Hoh <joerg@joerghoh.de>

Since June 2006, it has been maintained by Stephen Gran <sgran@debian.org>

deluser is Copyright (C) 2000 Roland Bauerschmidt <rb@debian.org> and based on the source code of adduser.

adduser is Copyright (C) 1997, 1998, 1999 Guy Maor <maor@debian.org>. adduser is Copyright (C) 1995 Ted Hajek <tedhajek@boombox.micro.umn.edu> with portions Copyright (C) 1994 Debian Association, Inc.

The examples directory has been contributed by John Zaitseff, and is GPL V2 as well.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL-2'.

1.316 vertx-core 3.9.4
1.316.1 Available under license:
The MIT License

Copyright (c) 2009 Carl Bystrm
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

JBoss, Home of Professional Open Source.
Copyright 2015 Red Hat, Inc., and individual contributors as indicated by the @author tags.

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

******************************************************************************
* Copyright (c) 1998-2010 AOL Inc.                                       *
*                                                                         *
* Licensed under the Apache License, Version 2.0 (the "License");        *
* you may not use this file except in compliance with the License.       *
* You may obtain a copy of the License at                                 *
*                                                                         *
* http://www.apache.org/licenses/LICENSE-2.0                             *
*                                                                         *
* Unless required by applicable law or agreed to in writing, software    *
* distributed under the License is distributed on an "AS IS" BASIS,       *
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. *
* See the License for the specific language governing permissions and    *
* limitations under the License.                                         *
1.317 free-type 2.8.1
1.317.1 Available under license:

The FreeType Project LICENSE
----------------------------
2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction
============

The FreeType Project is distributed in several archive packages;
some of them may contain, in addition to the FreeType font engine,
various tools and contributions which rely on, or relate to, the
FreeType Project.

This license applies to all files found in such packages, and
which do not fall under their own explicit license. The license
affects thus the FreeType font engine, the test programs,
documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG
(Independent JPEG Group) licenses, which all encourage inclusion
and use of free software in commercial and freeware products
alike. As a consequence, its main points are that:

- We don't promise that this software works. However, we will be
  interested in any kind of bug reports. (`as is' distribution)

- You can use this software for whatever you want, in parts or
  full form, without having to pay us. (`royalty-free' usage)

- You may not pretend that you wrote this software. If you use
  it, or only parts of it, in a program, you must acknowledge
  somewhere in your documentation that you have used the
  FreeType code. (`credits')

We specifically permit and encourage the inclusion of this
software, with or without modifications, in commercial products.
We disclaim all warranties covering The FreeType Project and
assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```plaintext
===
Portions of this software are copyright <year> The FreeType Project (www.freetype.org). All rights reserved.
===

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms
============

0. Definitions
--------------

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty
--------------
THE FREETYPE PROJECT IS PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file ("FTL.TXT") unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.

- Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: ’FreeType Project’, ’FreeType Engine’, ’FreeType library’, or ’FreeType Distribution’.
As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

- freetype@nongnu.org

  Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

- freetype-devel@nongnu.org

  Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

https://www.freetype.org

--- end of FTL.TXT ---

# Files that don't get a copyright, or which are taken from elsewhere.
#
# All lines in this file are patterns, including the comment lines; this
# means that e.g. `FTL.TXT' matches all files that have this string in
# the file name (including the path relative to the current directory,
# always starting with `./').
#
# Don't put empty lines into this file!
#
.gitignore
#
builds/unix/pkg.m4
#
docs/FTL.TXT
docs/GPLv2.TXT
#
include/freetype/internal/fhash.h
#
src/base/fthash.c
src/base/md5.c
src/base/md5.h
#
src/bdf/bdf.c
src/bdf/bdf.h
src/bdf/bdfdrivr.c
src/bdf/bdfdrivr.h
src/bdf/bdferror.h
src/bdf/bdflib.c
src/bdf/module.mk
src/bdf/README
src/bdf/rules.mk
#
src/pcf/module.mk
src/pcf/pcf.c
src/pcf/pcf.h
src/pcf/pcfdriver.c
src/pcf/pcfdriver.h
src/pcf/pcferror.h
src/pcf/pcfread.c
src/pcf/pcfread.h
src/pcf/pcfutil.c
src/pcf/pcfutil.h
src/pcf/README
src/pcf/rules.mk
#
src/gzip/adler32.c
src/gzip/infblock.c
src/gzip/infblock.h
src/gzip/infcodes.c
src/gzip/infcodes.h
src/gzip/inffixed.h
src/gzip/inflate.c
src/gzip/inftrees.c
src/gzip/inftrees.h
src/gzip/infutil.c
src/gzip/infutil.h
src/gzip/zconf.h
src/gzip/zlib.h
src/gzip/zutil.c
src/gzip/zutil.h
#
src/tools/apinames.c
src/tools/ftrandom/ftrandom.c
#
# EOF
The FreeType 2 font engine is copyrighted work and cannot be used legally without a software license. In order to make this project usable to a vast majority of developers, we distribute it under two mutually exclusive open-source licenses.

This means that *you* must choose *one* of the two licenses described below, then obey all its terms and conditions when using FreeType 2 in any of your projects or products.

- The FreeType License, found in the file `FTL.TXT', which is similar to the original BSD license *with* an advertising clause that forces you to explicitly cite the FreeType project in your product's documentation. All details are in the license file. This license is suited to products which don't use the GNU General Public License.

  Note that this license is compatible to the GNU General Public License version 3, but not version 2.

- The GNU General Public License version 2, found in `GPLv2.TXT' (any later version can be used also), for programs which already use the GPL. Note that the FTL is incompatible with GPLv2 due to its advertisement clause.

The contributed BDF and PCF drivers come with a license similar to that of the X Window System. It is compatible to the above two licenses (see file src/bdf/README and src/pcf/README). The same holds for the files `fhash.c' and `fhash.h'; their code was part of the BDF driver in earlier FreeType versions.

The gzip module uses the zlib license (see src/gzip/zlib.h) which too is compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds) is in the public domain.

--- end of LICENSE.TXT ---

**GNU GENERAL PUBLIC LICENSE**

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.
GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)
The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.
10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.
1.318.1 Available under license:

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty
percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.
3. Distribution Obligations.

3.1. Availability of Source Code.
Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.
The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.
When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly
granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any
distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING
NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER
CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF
SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES
FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR
MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH
PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION
OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING
FROM SUCH PARTY’S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH
LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF
INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT
APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of
commercial computer software, (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and commercial
computer software documentation, as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48
Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and
supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer
software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License
is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.
This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original
Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction’s conflict-of-
law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the
jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible
for costs, including, without limitation, court costs and reasonable attorneys’ fees and expenses. The application of
the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or
regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this
License. You agree that You alone are responsible for compliance with the United States export administration
regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise
make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising,
directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the
original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do
not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the
sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
1.319 shiro-crypto-core 1.7.1

1.319.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability.
incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.320 tar 1.29b 2ubuntu0.2

1.320.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—\textcolor{red}{to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.}

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of
software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited
permission to run the unmodified Program. The output from running a
covered work is covered by this License only if the output, given its
content, constitutes a covered work. This License acknowledges your
rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not
convey, without conditions so long as your license otherwise remains
in force. You may convey covered works to others for the sole purpose
of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
not control copyright. Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under
the conditions stated below. Sublicensing is not allowed; section 10
makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article
11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such
measures.

When you convey a covered work, you waive any legal power to forbid
circumvention of technological measures to the extent such circumvention
is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or
modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of
technological measures.


You may convey verbatim copies of the Program's source code as you
receive it, in any medium, provided that you conspicuously and
appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any
non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.
5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium
customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you
must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.
Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to
sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.


If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.
THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program>  Copyright (C) <year>  <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read

1.321 libaio 0.3.110-5ubuntu0.1
1.321.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is
modified by someone else and passed on, the recipients should know
that what they have is not the original version, so that the original
author's reputation will not be affected by problems that might be
introduced by others.

Finally, software patents pose a constant threat to the existence of
any free program. We wish to make sure that a company cannot
effectively restrict the users of a free program by obtaining a
restrictive license from a patent holder. Therefore, we insist that
any patent license obtained for a version of the library must be
consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the
ordinary GNU General Public License. This license, the GNU Lesser
General Public License, applies to certain designated libraries, and
is quite different from the ordinary General Public License. We use
this license for certain libraries in order to permit linking those
libraries into non-free programs.

When a program is linked with a library, whether statically or using
a shared library, the combination of the two is legally speaking a
combined work, a derivative of the original library. The ordinary
General Public License therefore permits such linking only if the
entire combination fits its criteria of freedom. The Lesser General
Public License permits more lax criteria for linking other code with
the library.

We call this license the "Lesser" General Public License because it
does Less to protect the user's freedom than the ordinary General
Public License. It also provides other free software developers Less
of an advantage over competing non-free programs. These disadvantages
are the reason we use the ordinary General Public License for many
libraries. However, the Lesser license provides advantages in certain
special circumstances.

For example, on rare occasions, there may be a special need to
encourage the widest possible use of a certain library, so that it
becomes
a de-facto standard. To achieve this, non-free programs must be
allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this
case, there is little to gain by limiting the free library to free
software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free
programs enables a greater number of people to use a large body of
free software. For example, permission to use the GNU C Library in
non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library’s complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

      (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the
user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.
NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.

END OF TERMS AND CONDITIONS

^L

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms
of the ordinary General Public License).

To apply these terms, attach the following notices to the library.
It is safest to attach them to the start of each source file to most
effectively convey the exclusion of warranty; and each file should
have at least the "copyright" line and a pointer to where the full
notice is found.

<one line to give the library's name and a brief idea of what it
does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.322 spring-data-commons 2.1.5.RELEASE
1.322.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=======================================================================

To the extent any open source subcomponents are licensed under the EPL and/or other similar licenses that require the source code and/or modifications to source code to be made available (as would be noted above), you may obtain a copy of the source code corresponding to the binaries for such open source components and modifications thereto, if any, (the "Source Files"), by
downloading the Source Files from http://www.springsource.org/download,
or by sending a request, with your name and address to: VMware, Inc., 3401 Hillview
Avenue, Palo Alto, CA 94304, United States of America or email info@vmware.com. All
such requests should clearly specify: OPEN SOURCE FILES REQUEST, Attention General
Counsel. VMware shall mail a copy of the Source Files to you on a CD or equivalent
physical medium. This offer to obtain a copy of the Source Files is valid for three
years from the date you acquired this Software product.

Spring Data Commons 2.1.5
Copyright (c) [2010-2019] Pivotal Software, Inc.

This product is licensed to you under the Apache License, Version 2.0 (the "License).
You may not use this product except in compliance with the License.

This product may include a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for the these subcomponents is subject to the terms and
conditions of the subcomponent's license, as noted in the LICENSE file.

1.323 readline 8.0.4
1.323.1 Available under license:
The MIT License (MIT)

Copyright 2010 Christopher Williams. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

1.324 spring-plugin-metadata 1.2.0.RELEASE
**1.324.1 Available under license:**

Apache License, Version 2.0
http://www.apache.org/licenses/LICENSE-2.0
Copyright 2010 the original author or authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
See the License for the specific language governing permissions and
limitations under the License.

**1.325 bzip2 1.0.6 8.1ubuntu0.2**

**1.325.1 Available under license:**

--------------------------------------------------------------------------
This program, "bzip2", the associated library "libbzip2", and all
documentation, are copyright (C) 1996-2019 Julian R Seward. All
rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must
not claim that you wrote the original software. If you use this
software in a product, an acknowledgment in the product
documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must
not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote
products derived from this software without specific prior written
permission.
THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org
bzip2/libbzip2 version 1.0.8 of 13 July 2019

1.326 jersey-hk2 2.30.1
1.326.1 Available under license :

# Notice for Jersey
This content is produced and maintained by the Eclipse Jersey project.

* Project home: https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright
All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

**Angular JS, v1.6.6**
* License: MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

**aopalliance Version 1**
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

**Bean Validation API 2.0.2**
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
  * by the @authors tag.

**Bootstrap v3.3.7**
* License: MIT license (https://github.com/twbs/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

**Google Guava Version 18.0**
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

**javax.inject Version 1**
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

**Javassist Version 3.25.0-GA**
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

**Jackson JAX-RS Providers Version 2.9.9**
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

**jQuery v1.12.4**
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* License: MIT & GPL (http://www.opensource.org/licenses/mit-license.php &
  http://www.gnu.org/licenses/gpl.html)
* Project: http://www.pasella.it/projects/jquery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public
domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 7.2
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) 2005, 2008. All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology,
  Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
  http://www.w3.org/Consortium/Legal/
  # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial
2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS
3.1 If a Contributor Distributes the Program in any form, then:

   a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

   b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
      i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

      ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

      iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

      iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

   a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

   b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION
Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY
EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its...
Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language.
(Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on
the Program, the distribution of the whole must be on the terms of this
License, whose permissions for other licensees extend to the entire
whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of a
storage or distribution medium does not bring the other work under the
scope of this License.

3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable
source code, which must be distributed under the terms of Sections 1
and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three
years, to give any third party, for a charge no more than your cost
of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to
distribute corresponding source code. (This alternative is allowed
only for noncommercial distribution and only if you received the
program in object code or executable form with such an offer, in
accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source code
means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to control
compilation and installation of the executable. However, as a special
exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement from your version.

1.327 shiro-config-core 1.7.1

1.327.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabut's publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.328 rdma-core 17.1-1ubuntu0.2

1.328.1 Available under license:

Copyright (c) 2013. Intel Corporation. All rights reserved.
Copyright (c) 2007. QLogic Corp. All rights reserved.
Copyright (c) 2005. PathScale, Inc. All rights reserved.

This software is available to you under a choice of one of two licenses. You may choose to be licensed under the terms of the GNU General Public License (GPL) Version 2, available from the file COPYING in the main directory of this source tree, or the OpenIB.org BSD license below:

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Patent licenses, if any, provided herein do not apply to combinations of this program with other software, or any other product whatsoever.

OpenIB.org BSD license (MIT variant)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: rdma-core
Upstream-Contact: Doug Ledford <dledford@redhat.com>, Leon Romanovsky <Leon@kernel.org>
Source: https://github.com/linux-rdma/rdma-core
Files: *
Copyright: disclaimed
License: BSD-MIT or GPL-2

Files: debian/*
Copyright: 2008, Genome Research Ltd
    2014, Ana Beatriz Guerrero Lopez <ana@debian.org>
    2015-2016, Jason Gunthorpe <jgunthorpe@obsidianresearch.com>
    2016-2017, Benjamin Drung <benjamin.drung@profitbricks.com>
    2016-2017, Talat Batheesh <talatb@mellanox.com>
License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
.
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.
.
You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
.
On Debian systems, the full text of the GNU General Public License
version 2 can be found in the file `/usr/share/common-licenses/GPL-2`.

Files: CMakeLists.txt
License: BSD-MIT or GPL-2

Files: buildlib/*
    2016-2017 Mellanox Technologies, Inc
License: BSD-MIT or GPL-2

Files: buildlib/fixup-include/rdma-vmw_pvrDMA-abi.h
Copyright: 2012-2016 VMware, Inc.
License: BSD-2-Clause or GPL-2

Files: buildlib/fixup-include/stdatomic.h
Copyright: 2011 Ed Schouten <ed@FreeBSD.org>
    David Chisnall <theraven@FreeBSD.org>
License: BSD-2-Clause

Files: ccan/*
License: BSD-MIT or GPL-2

Files: libibverbs/man/*
    libibverbs/neigh.h
    libibverbs/nl1_compat.h
    libibverbs/neigh.c
Copyright: disclaimed
License: BSD-2-clause

Files: librdmacm/*
    2005, Ammasso, Inc.
    2005, Voltaire Inc.
    2014-2015, Mellanox Technologies LTD.
License: BSD-MIT or GPL-2

Files: librdmacm/examples/cmtime.c
    librdmacm/examples/rcopy.c
    librdmacm/examples/rdma_client.c
    librdmacm/examples/rdma_server.c
    librdmacm/examples/rdma_xclient.c
    librdmacm/examples/rdma_xserver.c
    librdmacm/examples/riostream.c
    librdmacm/examples/rstream.c
    librdmacm/examples/udpong.c
    2014-2015, Mellanox Technologies LTD.
License: BSD-MIT

Files: librdmacm/docs/rsocket
Copyright: disclaimed
License: BSD-2-clause

Files: librdmacm/man/*
Copyright: disclaimed
License: BSD-2-clause

Files: providers/bnxt_re/*
Copyright: 2015-2017, Broadcom Limited and/or its subsidiaries
License: BSD-2-clause or GPL-2

Files: providers/cxgb3/*
    providers/cxgb4/*
Copyright: 2003-2016, Chelsio Communications, Inc.
License: BSD-MIT or GPL-2

Files: providers/hfi1verbs/*
Copyright: 2005 PathScale, Inc.
2006-2009 QLogic Corporation
2015 Intel Corporation
License: BSD-3-clause or GPL-2

Files: providers/hns/*
Copyright: 2016, Hisilicon Limited.
License: BSD-MIT or GPL-2

Files: providers/i40iw/*
Copyright: 2015-2016, Intel Corporation.
License: BSD-MIT or GPL-2

Files: providers/ipathverbs/*
Copyright: 2006-2010, QLogic Corp.
2005, PathScale, Inc.
2013, Intel Corporation
License: BSD-MIT or GPL-2

Files: providers/mlx4/*
Copyright: 2004-2005, Topspin Communications.
2005-2007, Cisco, Inc.
2005-2017, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: providers/mlx5/*
Copyright: 2010-2017, Mellanox Technologies, Inc.
License: BSD-MIT or GPL-2

Files: providers/mlx5/man/*.3
providers/mlx5/man/*.7
Copyright: disclaimed
License: BSD-MIT

Files: providers/mthca/*
Copyright: 2004-2005, Topspin Communications.
2005, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: providers/nes/*
Copyright: 2006-2010, Intel Corporation.
License: BSD-MIT or GPL-2

Files: providers/ocrdma/*
Copyright: 2008-2013, Emulex.
License: BSD-2-clause or GPL-2
Files: providers/qedr/*
Copyright: 2015-2016, QLogic Corporation.
License: BSD-MIT or GPL-2

Files: providers/rxe/*
Copyright: 2009-2011, System Fabric Works, Inc.
  2009-2011, Mellanox Technologies Ltd.
  2005, PathScale, Inc.
License: BSD-MIT or GPL-2

Files: providers/vmw_pvrdma/*
Copyright: 2012-2016 VMware, Inc.
License: BSD-2-clause or GPL-2

Files: rdma-ndd/*
Copyright: 2004-2016, Intel Corporation.
License: BSD-MIT or GPL-2

Files: redhat/*
Copyright: 1996-2013, Red Hat, Inc.
License: GPL-2

Files: srp_daemon/*
Copyright: 2005, Topspin Communications.
  2006, Cisco Systems, Inc.
  2006, Mellanox Technologies Ltd.
License: BSD-MIT or GPL-2

Files: srp_daemon/srp_daemon.1.in
Copyright: 2006 Mellanox Technologies.
License: CPL-1.0 or BSD-2-clause or GPL-2

Files: srp_daemon/srpd.in
  srp_daemon/ibsrpd.in
Copyright: disclaimed
License: BSD-2-clause

Files: util/udma_barrier.h
Copyright: 2005 Topspin Communications.
License: BSD-MIT or GPL-2

License: BSD-MIT
  OpenIB.org BSD license (MIT variant)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

License: BSD-2-clause

OpenIB.org BSD license (FreeBSD Variant)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: BSD-3-clause

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer.
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.
* Neither the name of Intel Corporation nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

License: GPL-2
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; version 2 of the License.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

On Debian systems, the full text of the GNU General Public License
version 2 can be found in the file `/usr/share/common-licenses/GPL-2`.

License: CC0
The laws of most jurisdictions throughout the world automatically confer
exclusive Copyright and Related Rights (defined below) upon the creator and
subsequent owner(s) (each and all, an "owner") of an original work of
authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the
purpose of contributing to a commons of creative, cultural and scientific works
(“Commons”) that the public can reliably and without fear of later claims of
infringement build upon, modify, incorporate in other works, reuse and
redistribute as freely as possible in any form whatsoever and for any purposes,
including without limitation commercial purposes. These owners may contribute
to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work; moral rights retained by the original author(s) and/or performer(s); publicity and privacy rights pertaining to a person's image or likeness depicted in a Work; rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below; rights protecting the extraction, dissemination, use and reuse of data in a Work; database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be
judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

License: MIT

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

License: CPL-1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution ‘originates’ from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program” means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of
such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another

Open Source Used In IOx-FogDirector-AC18 1.17.0 3599
language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

   Gnomovision version 69, Copyright (C) year name of author
   Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
   This is free software, and you are welcome to redistribute it
   under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

   Yoyodyne, Inc., hereby disclaims all copyright interest in the program
   `Gnomovision' (which makes passes at compilers) written by James Hacker.

   <signature of Ty Coon>, 1 April 1989
   Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.
For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the “Affirmer”), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights (“Copyright and Related Rights”). Copyright and Related Rights include, but are not limited to, the following:

   the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work; moral rights retained by the original author(s) and/or performer(s); publicity and privacy rights pertaining to a person's image or likeness depicted in a Work; rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below; rights protecting the extraction, dissemination, use and reuse of data in a Work; database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non
transferable, non sublicensable, non exclusive, irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

OpenIB.org BSD license (FreeBSD Variant)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Default Dual License

Unless otherwise stated this software is available to you under a choice of
one of two licenses. You may choose to be licensed under the terms of the
OpenIB.org BSD (MIT variant) license (see COPYING.BSD_MIT) or the GNU General
Public License (GPL) Version 2 (see COPYING.GPL2), both included in this
package.

Files marked 'See COPYING file' are licensed under the above Dual License.

# Other Options

Individual source files may use a license different from the above Default Dual
License. If a license is declared in the file then it supersedes the Default
License.

If a directory contains a COPYING file then the License from that file becomes
the Default License for files in that directory and below.

# Copyright Holders

Refer to individual files for information on the copyright holders.

# License Catalog (Informative, Non Binding)

## Utilities

Utility source code that may be linked into any binary are available under
several licenses:

- MIT license (see ccan/LICENSE.MIT)
- Creative Commons CC0 1.0 Universal License (see ccan/LICENSE.CC0)

## Providers

The following providers use a different license than the Default Dual
License. Refer to files in each directory for details.

hfi1verbs
: Dual License: GPLv2 or Intel 3 clause BSD license

ipathverbs
### Libraries

All library compilable source code (.c and .h files) are available under the Default Dual License.

Unmarked ancillary files may be available under a Dual License: GPLv2 or OpenIB.org BSD (FreeBSD variant).

### Tools (iwpmd, srp_daemon, ibacm)

All compilable source code (.c and .h files) are available under the Default Dual License.

Unmarked ancillary files may be available under a Dual License: GPLv2 or OpenIB.org BSD (FreeBSD variant).

srp_daemon/srp_daemon/srp_daemon.sh: Any one of the GPLv2, a 2 clause BSD license or the CPLv1.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

1.329 netty 4.1.49.Final

1.329.1 Available under license:

Copyright (c) 2003-2008 Yuta Mori All Rights Reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/
TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner.
or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{ }
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2009 William Kinney

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This copy of Aalto XML processor is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/" (BSD License: http://www.opensource.org/licenses/bsd-license)

Copyright (c) 2011, Joe Walnes, Aslak Hellesy and contributors
All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Webbit nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Netty Project
=================

Please visit the Netty web site for more information:

* https://netty.io/

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:
  * license/LICENSE.jsr166y.txt (Public Domain)
* HOMEPAGE:
  * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
  * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:
  * license/LICENSE.base64.txt (Public Domain)
* HOMEPAGE:
  * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:
  * license/LICENSE.webbit.txt (BSD License)
* HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
* HOMEPAGE:
  * http://www.slf4j.org/

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

* NOTICE:
This product contains a modified portion of 'jbzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

* LICENSE:
  * license/LICENSE.jbzip2.txt (MIT License)
  * https://code.google.com/p/jbzip2/

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

* LICENSE:
  * license/LICENSE.libdivsufsort.txt (MIT License)
  * https://github.com/y-256/libdivsufsort

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

* LICENSE:
  * license/LICENSE.jctools.txt (ASL2 License)
  * https://github.com/JCTools/JCTools

This product optionally depends on 'JZlib', a re-implementation of zlib in pure Java, which can be obtained at:

* LICENSE:
  * license/LICENSE.jzlib.txt (BSD style License)
  * http://www.jcraft.com/jzlib/

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

* LICENSE:
  * license/LICENSE.compress-lzf.txt (Apache License 2.0)
  * https://github.com/ning/compress

This product optionally depends on 'lz4', a LZ4 Java compression
and decompression library written by Adrien Grand. It can be obtained at:

* LICENSE:
  * license/LICENSE.lz4.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jpountz/lz4-java

This product optionally depends on 'Izma-java', a LZMA Java compression and decompression library, which can be obtained at:

* LICENSE:
  * license/LICENSE.lzma-java.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jponge/lzma-java

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

* LICENSE:
  * license/LICENSE.jfastlz.txt (MIT License)
* HOMEPAGE:
  * https://code.google.com/p/jfastlz/

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data interchange format, which can be obtained at:

* LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/protobuf

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

* LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
* HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

* LICENSE:
  * license/LICENSE.snappy.txt (New BSD License)
* HOMEPAGE:
  * https://github.com/google/snappy
This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-marshalling.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/jboss-remoting/jboss-marshalling

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.caliper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/google/caliper

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
* HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

* LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
* HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

* LICENSE:
  * license/LICENSE.alto-xml.txt (Apache License 2.0)
* HOMEPAGE:
  * http://wiki.fasterxml.com/AaltoHome

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

* LICENSE:
  * license/LICENSE.hpack.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/twitter/hpack
This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Cory Benfield. It can be obtained at:

* LICENSE:
  * license/LICENSE.hyper-hpack.txt (MIT License)
* HOMEPAGE:
  * https://github.com/python-hyper/hpack/

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Tatsuhiro Tsujikawa. It can be obtained at:

* LICENSE:
  * license/LICENSE.nghttp2-hpack.txt (MIT License)
* HOMEPAGE:
  * https://github.com/nghttp2/nghttp2/

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

* LICENSE:
  * license/LICENSE.commons-lang.txt (Apache License 2.0)
* HOMEPAGE:
  * https://commons.apache.org/proper/commons-lang/

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:
  * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)
* HOMEPAGE:
  * https://github.com/takari/maven-wrapper

This product contains the dnsinfo.h header file, that provides a way to retrieve the system DNS configuration on MacOS. This private header is also used by Apple's open source mDNSResponder (https://opensource.apple.com/tarballs/mDNSResponder/).

* LICENSE:
  * license/LICENSE.dnsinfo.txt (Apache License 2.0)
* HOMEPAGE:

The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright 2011, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
  distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Harmony

Copyright 2006, 2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Copyright (c) 2010-2011 Matthew J. Francis and Contributors of the jbzip2 Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright 2009-2010 Ning, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuate of all present and future rights under copyright law, whether vested or contingent, in the
Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate.
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY. WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT,
INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA,
OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
/*
* Copyright (c) 2004-2007 QOS.ch
* All rights reserved.
* Permission is hereby granted, free of charge, to any person obtaining
* a copy of this software and associated documentation files (the
* "Software"), to deal in the Software without restriction, including
* without limitation the rights to use, copy, modify, merge, publish,
* distribute, sublicense, and/or sell copies of the Software, and to
* permit persons to whom the Software is furnished to do so, subject to
* the following conditions:
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/
/*
* Copyright (c) 2004-2006, 2008, 2009, 2011 Apple Inc. All rights reserved.
* @APPLE_LICENSE_HEADER_START@
* This file contains Original Code and/or Modifications of Original Code
* as defined in and that are subject to the Apple Public Source License
* Version 2.0 (the 'License'). You may not use this file except in
* compliance with the License. Please obtain a copy of the License at

* http://www.opensource.apple.com/apsl/ and read it before using this
* file.
*
* The Original Code and all software distributed under the License are
distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT.
Please see the License for the specific language governing rights and
limitations under the License.
*
* @APPLE_LICENSE_HEADER_END@
*/
Protocol Buffers - Google's data interchange format
Copyright 2013 Google Inc.  All rights reserved.
https://developers.google.com/protocol-buffers/

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following disclaimer
  in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
    Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTicular PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.330 systemd 237-3ubuntu10.53
1.330.1 Available under license:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices...
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your
cost of physically performing source distribution, a complete
machine-readable copy of the corresponding source code, to be
distributed under the terms of Sections 1 and 2 above on a medium
customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer
to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable. However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

5. You are not required to accept this License, since you have not
signed it. However, nothing else grants you permission to modify or
distribute the Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions. You may not impose any further
restrictions on the recipients’ exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License. If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.331 libiscsi7 1.17.0-1.1

1.331.1 Available under license:

Libiscsi components fall under two separate licences

The lib and include directories

The iscsi client library itself, i.e. the lib and include directories, is licenced under GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

The library also contains one file: lib/md5.c that is under Public Domain.

A copy of LGPL 2.1 is included in the file LICENCE-LGPL-2.1.txt but can also be downloaded from http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

This is the licence that applies to the libiscsi library and its use.

src/ld_iscsi.c

This LD_PRELOAD utility is licenced under GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

The src, examples and test-tool directories EXCEPT src/ld_iscsi.c

The utility and example applications using this library, i.e. the src and the examples directories, are licenced under the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

These are executable applications that link with the libiscsi library.

A copy of GPL 2 is included in the file LICENCE/GPL-2.txt but can also be downloaded from http://www.gnu.org/licenses/old-licenses/gpl-2.0.html

This licence applies only to the source code for the provided applications that link to and use libiscsi.
To avoid any confusion, every source file also contains a licence boilerplate.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free
library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) The modified work must itself be a software library.

   b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

   c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

   d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

   (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to
5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:
a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.
7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

   a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

   b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you
may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
Provide the program "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the program is with you. Should the program prove defective, you assume the cost of all necessary servicing, repair or correction.

12. In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may modify and/or redistribute the program as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.

End of Terms and Conditions

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

1.332 commons-chain 1.2
1.332.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Chain
Copyright 2003-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.333 netty-codec 4.1.49.Final

1.333.1 Available under license :

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/CharSequenceValueConverter.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/ValueConverter.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/DefaultHeadersImpl.java

No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2015 The Netty Project
* 
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* 
* http://www.apache.org/licenses/LICENSE-2.0
* 
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/protobuf/ProtobufEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/protobuf/ProtobufEncoderNano.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/protobuf/ProtobufDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/ProtocolDetectionState.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/UnsupportedValueConverter.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/protobuf/ProtobufVarint32FrameDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/headers/HeadersUtils.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2019 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Lz4XXHash32.java

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ http://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4.1.49-final-sources-1-jar/META-INF/maven/io.netty/netty-codec/pom.xml
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2014 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License, version 2.0 (the
* "License"); you may not use this file except in compliance with the License. You may obtain a
* copy of the License at:
* *
* http://www.apache.org/licenses/LICENSE-2.0
* *
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4.1.49-final-sources-1-jar/io/netty/handler/codec/DefaultHeaders.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4.1.49-final-sources-1-jar/io/netty/handler/codec/Headers.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4.1.49-final-sources-1-jar/io/netty/handler/codec/EmptyHeaders.java
No license file was found, but licenses were detected in source scan.

/ *
* Copyright 2012 The Netty Project
* *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/MessageAggregator.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/CompatibleMarshallingDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/MarshallerProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/LimitingByteInput.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/PrematureChannelClosureException.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/MessageToMessageDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/DefaultUnmarshallProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/ThreadLocalMarshallerProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/DefaultUnmarshallProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/ThreadLocalUnmarshallProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/CompatbleMarshallingEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/SerializationObjectEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/CorruptedFrameException.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/LimitingByteInput.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/DefaultUnmarshallProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/ThreadLocalUnmarshallProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/CompatbleMarshallingEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/SerializationObjectEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/sharing/CorruptedFrameException.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2013 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 *   http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/MarshallingDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/ContextBoundUnmarshallerProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/JZlibDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/serialization/ObjectEncoderOutputStream.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/UnmarshallerProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/marshalling/UnmarshallarProvider.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/serialization/CompactObjectInputstream.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Crc32c.java
No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2012 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/*
 * Written by Robert Harder and released to the public domain, as explained at
 * http://creativecommons.org/licenses/publicdomain
 */

/**
 * Enumeration of supported Base64 dialects.
 * 
 * The internal lookup tables in this class has been derived from
 * <a href="http://iharder.sourceforge.net/current/java/base64/">Robert Harder's Public Domain
 * Base64 Encoder/Decoder</a>.
 */

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-
  jar/io/netty/handler/codec/base64/Base64Dialect.java

No license file was found, but licenses were detected in source scan.

/*
 * Copyright 2016 The Netty Project
 * 
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 * 
 * http://www.apache.org/licenses/LICENSE-2.0
 * 
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/string/LineEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/DatagramPacketDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/string/LineSeparator.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/ByteBufChecksum.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/DateFormatter.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/CodecOutputList.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/MessageAggregationException.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2Encoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Lz4FrameDecoder.java

No license file was found, but licenses were detected in source scan.

*/

* Copyright 2014 The Netty Project
 *
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/MessageAggregationException.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2Encoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Lz4FrameDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2HuffmanAllocator.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/FastLzFrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2MTFAndRLE2StageEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Lz4FrameEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/LzfDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2MTFAndRLE2StageDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Crc32.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2BlockCompressor.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2BlockDecompressor.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Lz4Constants.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/LzfEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2Decoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/FastLz.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageDecoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Bzip2HuffmanStageEncoder.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/compression/Crc32.java
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1-jar/io/netty/handler/codec/json/package-info.java
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */

/**
 * A decoder that splits the received {@link ByteBuf}s dynamically by the
 * value of the length field in the message. It is particularly useful when you
 * decode a binary message which has an integer header field that represents the
 * length of the message body or the whole message.
 * <p>
 * [{@link LengthFieldBasedFrameDecoder} has many configuration parameters so
 * that it can decode any message with a length field, which is often seen in
 * proprietary client-server protocols. Here are some example that will give
 * you the basic idea on which option does what.
 */

/*<p>
 * [{@link LengthFieldBasedFrameDecoder} has many configuration parameters so
 * that it can decode any message with a length field, which is often seen in
 * proprietary client-server protocols. Here are some example that will give
 * you the basic idea on which option does what.
 *<p>
 * 2 bytes length field at offset 0, do not strip header</p>
 *
 * The value of the length field in this example is <tt>12 (0x0C)</tt> which
 * represents the length of "HELLO, WORLD". By default, the decoder assumes
 * that the length field represents the number of the bytes that follows the
 * length field. Therefore, it can be decoded with the simplistic parameter
 * combination.
 */

/*<p>
 * lengthFieldOffset = <b>0</b>
 * lengthFieldLength = <b>2</b>
 * lengthAdjustment = 0
 * initialBytesToStrip = 0 (= do not strip header)
 *
 * BEFORE DECODE (14 bytes) AFTER DECODE (14 bytes)
 * | Length | Actual Content |----->| Length | Actual Content |
 */
Because we can get the length of the content by calling
[@link ByteBuf#readableBytes()], you might want to strip the length
field by specifying <tt>initialBytesToStrip</tt>. In this example, we
specified <tt>2</tt>, that is same with the length of the length field, to
strip the first two bytes.

```
lengthFieldOffset   = 0
lengthFieldLength   = 2
lengthAdjustment    = 0
initialBytesToStrip = 2
```

BEFORE DECODE (14 bytes)   AFTER DECODE (12 bytes)
```
| Length | Actual Content |
+--------+----------------+
| 0x000C | "HELLO, WORLD" |
```

BEFORE DECODE (14 bytes)   AFTER DECODE (14 bytes)
```
| Length | Actual Content |
+--------+----------------+
| 0x000E | "HELLO, WORLD" |
```

3 bytes length field at the end of 5 bytes header, do not strip header

In most cases, the length field represents the length of the message body
only, as shown in the previous examples. However, in some protocols, the
length field represents the length of the whole message, including the
message header. In such a case, we specify a non-zero
lengthAdjustment. Because the length value in this example message
is always greater than the body length by <tt>2</tt>, we specify <tt>-2</tt>
as lengthAdjustment for compensation.

```
lengthFieldOffset   = 0
lengthFieldLength   = 2
lengthAdjustment    = -2
initialBytesToStrip = 0
```

BEFORE DECODE (14 bytes)   AFTER DECODE (14 bytes)
```
| Length | Actual Content |
+--------+----------------+
| 0x000E | "HELLO, WORLD" |
```

3 bytes length field at the end of 5 bytes header, do not strip header

The following message is a simple variation of the first example. An extra header value is prepended to the message. `<tt>lengthAdjustment</tt>` is zero again because the decoder always takes the length of the prepended data into account during frame length calculation.

```
* <pre>
* <b>lengthFieldOffset</b>   = <b>2</b> (= the length of Header 1)
* <b>lengthFieldLength</b>   = <b>3</b>
* lengthAdjustment    = 0
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* | Header 1 |  Length  | Actual Content |----->| Header 1 |  Length  | Actual Content |
* | 0xCAFE  | 0x00000C | "HELLO, WORLD" |      | 0xCAFE  | 0x00000C | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>
```

This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive `<tt>lengthAdjustment</tt>` so that the decoder counts the extra header into the frame length calculation.

```
* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* |  Length  | Header 1 | Actual Content |----->|  Length  | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE  | "HELLO, WORLD" |      | 0x00000C | 0xCAFE  | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>
```

This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* <h3>2 bytes length field at the beginning of 5 bytes header, do not strip header</h3>
* This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive `<tt>lengthAdjustment</tt>` so that the decoder counts the extra header into the frame length calculation.

```
* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* |  Length  | Header 1 | Actual Content |----->|  Length  | Header 1 | Actual Content |
* | 0x00000C | 0xCAFE  | "HELLO, WORLD" |      | 0x00000C | 0xCAFE  | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>
```

This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>
* This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive `<tt>lengthAdjustment</tt>` so that the decoder counts the extra header into the frame length calculation.

```
* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* | Header 1 |  Length  | Actual Content |----->| Header 1 |  Length  | Actual Content |
* | 0xCAFE  | 0x00000C | "HELLO, WORLD" |      | 0xCAFE  | 0x00000C | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>

This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header, strip the first header field and the length field</h3>
* This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* <h3>3 bytes length field at the beginning of 5 bytes header, do not strip header</h3>
* This is an advanced example that shows the case where there is an extra header between the length field and the message body. You have to specify a positive `<tt>lengthAdjustment</tt>` so that the decoder counts the extra header into the frame length calculation.

```
* <pre>
* lengthFieldOffset   = 0
* lengthFieldLength   = 3
* <b>lengthAdjustment</b>    = <b>2</b> (= the length of Header 1)
* initialBytesToStrip = 0
*
* BEFORE DECODE (17 bytes)                      AFTER DECODE (17 bytes)
* +----------+----------+----------------+      +----------+----------+----------------+
* | Header 1 |  Length  | Actual Content |----->| Header 1 |  Length  | Actual Content |
* | 0xCAFE  | 0x00000C | "HELLO, WORLD" |      | 0xCAFE  | 0x00000C | "HELLO, WORLD" |
* +----------+----------+----------------+      +----------+----------+----------------+
* </pre>

This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* <h3>2 bytes length field at offset 1 in the middle of 4 bytes header, strip the first header field and the length field</h3>
* This is a combination of all the examples above. There are the prepended header before the length field and the extra header after the length field. The prepended header affects the `<tt>lengthFieldOffset</tt>` and the extra header affects the `<tt>lengthAdjustment</tt>`. We also specified a non-zero `<tt>initialBytesToStrip</tt>` to strip the length field and the prepended header from the frame. If you don't want to strip the prepended header, you could specify `<tt>0</tt>` for `<tt>initialBytesToSkip</tt>`.

```
* <pre>
* lengthFieldOffset  = 1 (= the length of HDR1)
* lengthFieldLength  = 2
* <b>&lt;b&gt;lengthAdjustment&lt;/b&gt; = &lt;b&gt;1&lt;/b&gt; (= the length of HDR2)
* <b>&lt;b&gt;initialBytesToStrip&lt;/b&gt; = &lt;b&gt;3&lt;/b&gt; (= the length of HDR1 + LEN)

* BEFORE DECODE (16 bytes)          AFTER DECODE (13 bytes)
* +--------------------+-----------------+      +--------------------+-----------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x000C | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +--------------------+-----------------+      +--------------------+-----------------+
* </pre>

* 2 bytes length field at offset 1 in the middle of 4 bytes header,
*  strip the first header field and the length field, the length field
*  represents the length of the whole message</h3>

* Let's give another twist to the previous example. The only difference from
*  the previous example is that the length field represents the length of the
*  whole message instead of the message body, just like the third example.
*  We have to count the length of HDR1 and Length into &lt;tt&gt;lengthAdjustment&lt;/tt&gt;.
*  Please note that we don't need to take the length of HDR2 into account
*  because the length field already includes the whole header length.

* lengthFieldOffset  = 1
* lengthFieldLength  = 2
* <b>&lt;b&gt;lengthAdjustment&lt;/b&gt; = &lt;b&gt;-3&lt;/b&gt; (= the length of HDR1 + LEN, negative)
* <b>&lt;b&gt;initialBytesToStrip&lt;/b&gt; = &lt;b&gt;3&lt;/b&gt;

* BEFORE DECODE (16 bytes)          AFTER DECODE (13 bytes)
* +--------------------+-----------------+      +--------------------+-----------------+
* | HDR1 | Length | HDR2 | Actual Content |----->| HDR2 | Actual Content |
* | 0xCA | 0x0010 | 0xFE | "HELLO, WORLD" |      | 0xFE | "HELLO, WORLD" |
* +--------------------+-----------------+      +--------------------+-----------------+
* </pre>

* @see LengthFieldPrepender
*/

Found in path(s):
* /opt/ws_local/PERMITS_SQL/1068232267_1594320776.43/0/netty-codec-4-1-49-final-sources-1.jar/io/netty/handler/codec/LengthFieldBasedFrameDecoder.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2012 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
License for the specific language governing permissions and limitations
under the License.

Written by Robert Harder and released to the public domain, as explained at
http://creativecommons.org/licenses/publicdomain

Utility class for {@link ByteBuf} that encodes and decodes to and from
Base64 notation.

The encoding and decoding algorithm in this class has been derived from
Robert Harder's Public Domain
Base64 Encoder/Decoder.

Found in path(s):
/netty-codec-4.1.49-final-sources-1-jar/io/netty/handler/codec/base64/Base64.java

1.334 jersey 2.30.1
1.334.1 Available under license:

This content is produced and maintained by the Eclipse Jersey project.

Project home: https://projects.eclipse.org/projects/ee4j.jersey

Trademarks
Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Public License v. 2.0 which is available at
http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code
The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

Angular JS, v1.6.6
* License MIT (http://www.opensource.org/licenses/mit-license.php)
* Project: http://angularjs.org
* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1
* License: all the source code provided by AOP Alliance is Public Domain.
* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2
* License: Apache License, 2.0
* Project: http://beanvalidation.org/1.1/
* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
* by the @authors tag.

Bootstrap v3.3.7
* License: MIT license (https://github.com/twitter/bootstrap/blob/master/LICENSE)
* Project: http://getbootstrap.com
* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1
* License: Apache License, 2.0
* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA
* License: Apache License, 2.0
* Project: http://www.javassist.org/
* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.
Jackson JAX-RS Providers Version 2.9.9
* License: Apache License, 2.0
* Project: https://github.com/FasterXML/jackson-jaxrs-providers
* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4
* License: jquery.org/license
* Project: jquery.org
* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3
* Project: http://www.pasella.it/projects/jQuery/barcode
* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266
* License: CC0
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at http://creativecommons.org/publicdomain/zero/1.0/

KineticJS, v4.7.1
* License: MIT license (http://www.opensource.org/licenses/mit-license.php)
* Copyright: Eric Rowell

org.objectweb.asm Version 7.2
* License: Modified BSD (http://asm.objectweb.org/license.html)
* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0
* License: Apache License, 2.0
* Copyright (c) 2005, 2008. All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core
* License: Apache License, 2.0
* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents
* License: W3C License
* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
http://www.w3.org/ Consortium/Legal/
# Notice for Jersey Core Common module
This content is produced and maintained by the Eclipse Jersey project.
* https://projects.eclipse.org/projects/ee4j.jersey

## Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

## Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

## Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at http://www.eclipse.org/legal/epl-2.0. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at https://www.gnu.org/software/classpath/license.html.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

## Source Code

The project maintains the following source code repositories:

* https://github.com/eclipse-ee4j/jersey

## Third-party Content

Google Guava Version 18.0
* License: Apache License, 2.0
* Copyright: (C) 2009 The Guava Authors

JSR-166 Extension - JEP 266
* License: Creative Commons 1.0 (CC0)
* No copyright
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
# Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
   Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;
   where such changes and/or additions to the Program originate from
   and are Distributed by that particular Contributor. A Contribution
   "originates" from a Contributor if it was added to the Program by
   such Contributor itself or anyone acting on such Contributor's behalf.
   Contributions do not include changes or additions to the Program that
   are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which
are necessarily infringed by the use or sale of its Contribution alone
or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this
Agreement.

"Recipient" means anyone who receives the Program under this Agreement
or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other
form, that is based on (or derived from) the Program and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that
results from an addition to, deletion from, or modification of the
contents of the Program, including, for purposes of clarity any new file
in Source Code form that contains any contents of the Program. Modified
Works shall not include works that contain only declarations,
interfaces, types, classes, structures, or files of the Program solely
in each case in order to link to, bind by name, or subclass the Program
or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available
in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making
modifications, including but not limited to software source code,
documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

   a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

   b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

   c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

   d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

   e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's
receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (“notices”) contained within the Program from any copy of
the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement,
including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate
entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

---

## The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is
intended to guarantee your freedom to share and change free software--to
make sure the software is free for all its users. This General Public
License applies to most of the Free Software Foundation's software and
to any other program whose authors commit to using it. (Some other Free
Software Foundation software is covered by the GNU Library General
Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price.
Our General Public Licenses are designed to make sure that you have the
freedom to distribute copies of free software (and charge for this
service if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone
to deny you these rights or to ask you to surrender the rights. These
restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis
or for a fee, you must give the recipients all the rights that you have.
You must make sure that they, too, receive or can get the source code.
And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software patents.
We wish to avoid the danger that redistributors of a free program will
individually obtain patent licenses, in effect making the program
proprietary. To prevent this, we have made it clear that any patent must
be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a
notice placed by the copyright holder saying it may be distributed under
the terms of this General Public License. The "Program", below, refers
to any such program or work, and a "work based on the Program" means
either the Program or any derivative work under copyright law: that is
to say, a work containing the Program or a portion of it, either
verbatim or with modifications and/or translated into another language.
(Hereinafter, translation is included without limitation in the term
"modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of running
the Program is not restricted, and the output from the Program is
covered only if its contents constitute a work based on the Program
(independent of having been made by running the Program). Whether that
is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source
code as you receive it, in any medium, provided that you conspicuously
and appropriately publish on each copy an appropriate copyright notice
and disclaimer of warranty; keep intact all the notices that refer to
this License and to the absence of any warranty; and give any other
recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of
it, thus forming a work based on the Program, and copy and distribute
such modifications or work under the terms of Section 1 above, provided
that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices
stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in
whole or in part contains or is derived from the Program or any part
thereof, to be licensed as a whole at no charge to all third parties
under the terms of this License.

c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a notice
that there is no warranty (or else, saying that you provide a
warranty) and that users may redistribute the program under these
conditions, and telling the user how to view a copy of this License.
(Exception: if the Program itself is interactive but does not
normally print such an announcement, your work based on the Program
is not required to print an announcement.)
These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is
normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.
If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
11. Because the program is licensed free of charge, there is no warranty for the program, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holders and/or other parties provide the program "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the program is with you. Should the program prove defective, you assume the cost of all necessary servicing, repair or correction.

12. In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may modify and/or redistribute the program as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.

End of terms and conditions

How to apply these terms to your new programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `show w' and `show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

---

## CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.335 shiro-quartz 1.7.1

1.335.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based on initial ideas from Dr. Heinz Kabutz's publicly posted version available at http://www.javaspecialists.eu/archive/Issue015.html, with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source code for this product was copied for simplicity and to reduce dependencies from the source code developed by the Spring Framework Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.336 commons-compress 1.19

1.336.1 Available under license :

Apache Commons Compress
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (https://www.apache.org/).
The files in the package org.apache.commons.compress.archivers.sevenz were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/), which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.337 kerberos 1.16-2ubuntu0.2

1.337.1 Available under license :

[ NOTE: MIT has only incorporated the mechglue and spnego change, and
not the incremental propagation changes. The filenames are different
between the Sun and MIT sources. The actual MIT filenames appear in
the top-level README file. Original text of Sun's LICENSE file
follows. ]

Subject to the license set forth below, Sun Microsystems, Inc. donates
the attached files to MIT for the purpose of including these
modifications and additions in future versions of the Kerberos system.

Many of the files attached are subject to licenses issued by other
entities, including OpenVision, MIT, and FundsXpress. See the
individual files, and/or related Readme files, for these licenses.

In addition Sun requires that the license set forth below be
incorporated into any future version of the Kerberos system which
contains portions of the files attached. The following files must be
listed, in the top level Readme file, as being provided subject to such
license:

cmd/krb5/iprop/iprop.x
cmd/krb5/iprop/iprop_hdr.h
cmd/krb5/kadmin/server/ipropd_svc.c
cmd/krb5/kproplog/kproplog.c
cmd/krb5/slave/kpropd_rpc_c
lib/gss_mechs/mech_krb5/et/kdb5_err.c
lib/gss_mechs/mech_spnego/mach/gssapiP_spnego.h
lib/gss_mechs/mech_spnego/mach/spnego_mach.c
lib/krb5/kadm5/kadm_host_srv_names.c
lib/krb5/kdb/kdb_convert.c
lib/krb5/kdb/kdb_hdr.h
lib/krb5/kdb/kdb_log.c
lib/krb5/kdb/kdb_log.h
lib/libgss/g_accept_sec_context.c
lib/libgss/g_acquire_cred.c
lib/libgss/g_canon_name.c
lib/libgss/g_compare_name.c
lib/libgss/g_context_time.c
lib/libgss/g_delete_sec_context.c
lib/libgss/g_dsp_name.c
lib/libgss/g_dsp_status.c
lib/libgss/g_dup_name.c
lib/libgss/g_exp_sec_context.c
lib/libgss/g_export_name.c
lib/libgss/g_glue.c
lib/libgss/g_imp_name.c
lib/libgss/g_imp_sec_context.c
lib/libgss/g_init_sec_context.c
lib/libgss/g_initialize.c
lib/libgss/g_inquire_context.c
lib/libgss/g_inquire_cred.c
lib/libgss/g_inquire_names.c
lib/libgss/g_process_context.c
lib/libgss/g_rel_buffer.c
lib/libgss/g_rel_cred.c
lib/libgss/g_rel_name.c
lib/libgss/g_rel_oid_set.c
lib/libgss/g_seal.c
lib/libgss/g_sign.c
lib/libgss/g_store_cred.c
lib/libgss/g_unseal.c
lib/libgss/g_userok.c
lib/libgss/g_utils.c
lib/libgss/gs dân_pname_to_uid.c
uts/common/gssapi/include/gssapi_err_generic.h
uts/common/gssapi/include/mechglueP.h

Sun's License is as follows:

Copyright (c) 2004 Sun Microsystems, Inc.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Copyright (C) 1985-2019 by the Massachusetts Institute of Technology.

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Downloading of this software may constitute an export of cryptographic
software from the United States of America that is subject to the
United States Export Administration Regulations (EAR), 15 CFR 730-774.
Additional laws or regulations may apply. It is the responsibility of
the person or entity contemplating export to comply with all
applicable export laws and regulations, including obtaining any
required license from the U.S. government.
The U.S. government prohibits export of encryption source code to certain countries and individuals, including, but not limited to, the countries of Cuba, Iran, North Korea, Sudan, Syria, and residents and nationals of those countries.

Documentation components of this software distribution are licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License. (https://creativecommons.org/licenses/by-sa/3.0/)

Individual source code files are copyright MIT, Cygnus Support, Novell, OpenVision Technologies, Oracle, Red Hat, Sun Microsystems, FundsXpress, and others.

Project Athena, Athena, Athena MUSE, Discuss, Hesiod, Kerberos, Moira, and Zephyr are trademarks of the Massachusetts Institute of Technology (MIT). No commercial use of these trademarks may be made without prior written permission of MIT.

"Commercial use" means use of a name in a product or other for-profit manner. It does NOT prevent a commercial firm from referring to the MIT trademarks in order to convey information (although in doing so, recognition of their trademark status should be given).

The following copyright and permission notice applies to the OpenVision Kerberos Administration system located in "kadmin/create", "kadmin/dbutil", "kadmin/passwd", "kadmin/server", "lib/kadm5", and portions of "lib/rpc":


WARNING: Retrieving the OpenVision Kerberos Administration system source code, as described below, indicates your acceptance of the following terms. If you do not agree to the following terms, do not retrieve the OpenVision Kerberos administration system.

You may freely use and distribute the Source Code and Object Code compiled from it, with or without modification, but this Source Code is provided to you "AS IS" EXCLUSIVE OF ANY WARRANTY, INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY OTHER WARRANTY, WHETHER EXPRESS OR IMPLIED. IN NO EVENT WILL OPENVISION HAVE ANY LIABILITY FOR ANY LOST PROFITS, LOSS OF DATA OR COSTS OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, INCLUDING, WITHOUT LIMITATION, THOSE RESULTING FROM THE USE OF THE SOURCE
CODE, OR THE FAILURE OF THE SOURCE CODE TO PERFORM, OR FOR ANY OTHER REASON.

OpenVision retains all copyrights in the donated Source Code. OpenVision also retains copyright to derivative works of the Source Code, whether created by OpenVision or by a third party. The OpenVision copyright notice must be preserved if derivative works are made based on the donated Source Code.

OpenVision Technologies, Inc. has donated this Kerberos Administration system to MIT for inclusion in the standard Kerberos 5 distribution. This donation underscores our commitment to continuing Kerberos technology development and our gratitude for the valuable work which has been performed by MIT and the Kerberos community.

======================================================================
Portions contributed by Matt Crawford "crawdad@fnal.gov" were work performed at Fermi National Accelerator Laboratory, which is operated by Universities Research Association, Inc., under contract DE-AC02-76CHO3000 with the U.S. Department of Energy.

======================================================================
Portions of "src/lib/crypto" have the following copyright:

Copyright (C) 1998 by the FundsXpress, INC.

All rights reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of FundsXpress. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. FundsXpress makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

======================================================================

The implementation of the AES encryption algorithm in "src/lib/crypto/builtin/aes" has the following copyright:

Copyright (C) 2001, Dr Brian Gladman "brg@gladman.uk.net", Worcester, UK.
All rights reserved.

LICENSE TERMS

The free distribution and use of this software in both source and binary form is allowed (with or without changes) provided that:

1. distributions of this source code include the above copyright notice, this list of conditions and the following disclaimer;

2. distributions in binary form include the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other associated materials;

3. the copyright holder's name is not used to endorse products built using this software without specific written permission.

DISCLAIMER

This software is provided 'as is' with no explicit or implied warranties in respect of any properties, including, but not limited to, correctness and fitness for purpose.

======================================================================

Portions contributed by Red Hat, including the pre-authentication plug-in framework and the NSS crypto implementation, contain the following copyright:

Copyright (C) 2006 Red Hat, Inc.
Portions copyright (C) 2006 Massachusetts Institute of Technology
All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Red Hat, Inc., nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The bundled verto source code is subject to the following license:

Copyright 2011 Red Hat, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
The MS-KKDCP client implementation has the following copyright:

Copyright 2013, 2014 Red Hat, Inc.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
and the initial implementation of incremental propagation, including the following new or changed files:

```
include/iprop_hdr.h
ekadmin/server/ipropd_svc.c
lib/kdb/iprop.x
lib/kdb/kdb_convert.c
lib/kdb/kdb_log.c
lib/kdb/kdb_log.h
lib/krb5/error_tables/kdb5_err.et
kprop/kpropd_rpc.c
kprop/kproplog.c
```

are subject to the following license:

Copyright (C) 2004 Sun Microsystems, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================
Kerberos V5 includes documentation and software developed at the University of California at Berkeley, which includes this copyright notice:

Copyright (C) 1983 Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS
OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Portions contributed by Novell, Inc., including the LDAP database backend, are subject to the following license:

Copyright (C) 2004-2005, Novell, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The copyright holder's name is not used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================
Portions funded by Sandia National Laboratory and developed by the University of Michigan's Center for Information Technology Integration, including the PKINIT implementation, are subject to the following license:

COPYRIGHT (C) 2006-2007
THE REGENTS OF THE UNIVERSITY OF MICHIGAN
ALL RIGHTS RESERVED

Permission is granted to use, copy, create derivative works and redistribute this software and such derivative works for any purpose, so long as the name of The University of Michigan is not used in any advertising or publicity pertaining to the use of distribution of this software without specific, written prior authorization. If the above copyright notice or any other identification of the University of Michigan is included in any copy of any portion of this software, then the disclaimer below must also be included.

THIS SOFTWARE IS PROVIDED AS IS, WITHOUT REPRESENTATION FROM THE UNIVERSITY OF MICHIGAN AS TO ITS FITNESS FOR ANY PURPOSE, AND WITHOUT WARRANTY BY THE UNIVERSITY OF MICHIGAN OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE REGENTS OF THE UNIVERSITY OF MICHIGAN SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WITH RESPECT TO ANY CLAIM ARISING OUT OF OR IN CONNECTION WITH THE USE OF THE SOFTWARE, EVEN IF IT HAS BEEN OR IS HEREAFTER ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

======================================================================

The pkcs11.h file included in the PKINIT code has the following license:

Copyright 2006 g10 Code GmbH
Copyright 2006 Andreas Jellinghaus

This file is free software; as a special exception the author gives unlimited permission to copy and/or distribute it, with or without modifications, as long as this notice is preserved.

This file is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY, to the extent permitted by law; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

======================================================================
Portions contributed by Apple Inc. are subject to the following license:

Copyright 2004-2008 Apple Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Apple Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Apple Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

======================================================================

The implementations of UTF-8 string handling in src/util/support and src/lib/krb5/unicode are subject to the following copyright and permission notice:

The OpenLDAP Public License
Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions in source form must retain copyright statements and notices,

2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and

3. Redistributions must contain a verbatim copy of this
The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distribute verbatim copies of this document is granted.

======================================================================

Marked test programs in src/lib/krb5/krb have the following copyright:

Copyright (C) 2006 Kungliga Tekniska Hgskola
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of KTH nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY KTH AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL KTH OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The KCM Mach RPC definition file used on macOS has the following copyright:

Copyright (C) 2009 Kungliga Tekniska Hgskola
(Royal Institute of Technology, Stockholm, Sweden).
All rights reserved.

Portions Copyright (C) 2009 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the Institute nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Portions of the RPC implementation in src/lib/rpc and src/include/gssrpc have the following copyright and permission notice:

Copyright (C) 2010, Oracle America, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the "Oracle America, Inc." nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Copyright (C) 2006,2007,2009 NTT (Nippon Telegraph and Telephone Corporation). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY NTT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL NTT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

Copyright 2000 by Carnegie Mellon University

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all
copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Carnegie Mellon University not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

======================================================================
Copyright (C) 2002 Naval Research Laboratory (NRL/CCS)

Permission to use, copy, modify and distribute this software and its documentation is hereby granted, provided that both the copyright notice and this permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof.

NRL ALLOWS FREE USE OF THIS SOFTWARE IN ITS "AS IS" CONDITION AND DISCLAIMS ANY LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

======================================================================

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Cygnus Support makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

======================================================================
Copyright (C) 2006 Secure Endpoints Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

======================================================================
Portions of the implementation of the Fortuna-like PRNG are subject to
the following notice:

    Copyright (C) 2005 Marko Kreen
    All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above
   copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Copyright (C) 1994 by the University of Southern California

EXPORT OF THIS SOFTWARE from the United States of America may
require a specific license from the United States Government. It
is the responsibility of any person or organization
contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to copy, modify, and distribute
this software and its documentation in source and binary forms is
hereby granted, provided that any documentation or other materials
related to such distribution or use acknowledge that the software
was developed by the University of Southern California.

DISCLAIMER OF WARRANTY. THIS SOFTWARE IS PROVIDED "AS IS". The
University of Southern California MAKES NO REPRESENTATIONS OR
WARRANTIES, EXPRESS OR IMPLIED. By way of example, but not
limitation, the University of Southern California MAKES NO
REPRESENTATIONS OR WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY
PARTICULAR PURPOSE. The University of Southern California shall not
be held liable for any liability nor for any direct, indirect, or
consequential damages with respect to any claim by the user or
distributor of the ksu software.

======================================================================
Copyright (C) 1995
The President and Fellows of Harvard University

This code is derived from software contributed to Harvard by Jeremy
Rassen.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above
copyright notice, this list of conditions and the following
disclaimer.

2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================
Copyright (C) 2008 by the Massachusetts Institute of Technology.
Copyright 1995 by Richard P. Basch. All Rights Reserved.
Copyright 1995 by Lehman Brothers, Inc. All Rights Reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Richard P. Basch, Lehman Brothers and M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Richard P. Basch, Lehman Brothers and M.I.T. make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

======================================================================
The following notice applies to "src/lib/krb5/krb/strptime.c" and "src/include/k5-queue.h".

Copyright (C) 1997, 1998 The NetBSD Foundation, Inc.
All rights reserved.

This code was contributed to The NetBSD Foundation by Klaus Klein.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

4. Neither the name of The NetBSD Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================
The following notice applies to Unicode library files in "src/lib/krb5/unicode":

Copyright 1997, 1998, 1999 Computing Research Labs,
New Mexico State University

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE COMPUTING RESEARCH LAB OR
NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE
OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

The following notice applies to "src/util/support/strlcpy.c":

Copyright (C) 1998 Todd C. Miller "Todd.Miller@courtesan.com"

Permission to use, copy, modify, and distribute this software for
any purpose with or without fee is hereby granted, provided that
the above copyright notice and this permission notice appear in all
copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL
WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE
AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS
OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
NEGligence OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

======================================================================
The following notice applies to "src/util/profile/argv_parse.c" and "src/util/profile/argv_parse.h":

Copyright 1999 by Theodore Ts'o.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies. THE SOFTWARE IS PROVIDED "AS IS" AND THEODORE TS'O (THE AUTHOR) DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. (Isn't it sick that the U.S. culture of lawsuit-happy lawyers requires this kind of disclaimer?)

======================================================================

The following notice applies to SWIG-generated code in "src/util/profile/profile_tcl.c":

Copyright (C) 1999-2000, The University of Chicago

This file may be freely redistributed without license or fee provided this copyright message remains intact.

======================================================================

The following notice applies to portions of "src/lib/rpc" and "src/include/gssrpc":

Copyright (C) 2000 The Regents of the University of Michigan. All rights reserved.

Copyright (C) 2000 Dug Song "dugsong@UMICH.EDU". All rights reserved, all wrongs reversed.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.

3. Neither the name of the University nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT
OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

======================================================================

Implementations of the MD4 algorithm are subject to the following
notice:

Copyright (C) 1990, RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it
is identified as the "RSA Data Security, Inc. MD4 Message Digest
Algorithm" in all material mentioning or referencing this software
or this function.

License is also granted to make and use derivative works provided
that such works are identified as "derived from the RSA Data
Security, Inc. MD4 Message Digest Algorithm" in all material
mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.

======================================================================
Implementations of the MD5 algorithm are subject to the following notice:

Copyright (C) 1990, RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

======================================================================

The following notice applies to "src/lib/crypto/crypto_tests/t_mddriver.c":

Copyright (C) 1990-2, RSA Data Security, Inc. Created 1990. All rights reserved.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

======================================================================

Portions of "src/lib/krb5" are subject to the following notice:

Copyright (C) 1994 CyberSAFE Corporation.
All Rights Reserved.
Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. Furthermore if you modify this software you must label your software as modified software and not distribute it in such a fashion that it might be confused with the original M.I.T. software. Neither M.I.T., the Open Computing Security Group, nor CyberSAFE Corporation make any representations about the suitability of this software for any purpose. It is provided "as is” without express or implied warranty.

================================================================================

Portions contributed by PADL Software are subject to the following license:

Copyright (c) 2011, PADL Software Pty Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of PADL Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY PADL SOFTWARE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL PADL SOFTWARE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The bundled libev source code is subject to the following license:

All files in libev are Copyright (C)2007,2008,2009 Marc Alexander Lehmann.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Alternatively, the contents of this package may be used under the terms of the GNU General Public License ("GPL") version 2 or any later version, in which case the provisions of the GPL are applicable instead of the above. If you wish to allow the use of your version of this package only under the terms of the GPL and not to allow others to use your version of this file under the BSD
license, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL in this and the other files of this package. If you do not delete the provisions above, a recipient may use your version of this file under either the BSD or the GPL.

======================================================================

Files copied from the Intel AESNI Sample Library are subject to the following license:

Copyright (C) 2010, Intel Corporation All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The following notice applies to "src/ccapi/common/win/OldCC/autolock.hxx":

---

Open Source Used In IOx-FogDirector-AC18 1.17.0 3737
Copyright (C) 1998 by Danilo Almeida. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

======================================================================

The following notice applies to portions of
"src/plugins/preauth/spake/edwards25519.c" and
"src/plugins/preauth/spake/edwards25519_tables.h":

The MIT License (MIT)

Copyright (c) 2015-2016 the fiat-crypto authors (see the AUTHORS file).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

======================================================================

The following notice applies to portions of
"src/plugins/preauth/spake/edwards25519.c":

Copyright (c) 2015-2016, Google Inc.

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

.. _mitK5license:

MIT Kerberos License information
================================

.. toctree::
   :hidden:
   :hidden:

   copyright

.. include::  notice

Copyright
=========

Copyright |copy| 1985-2019 by the Massachusetts Institute of Technology and its contributors. All rights reserved.

See :ref:`mitK5license` for additional copyright and license
1.338 init-system-helpers 1.51

1.338.1 Available under license:

Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/

Files: *
Copyright: 2013 Michael Stapelberg
License: BSD-3-clause

Files: debian/*
Copyright: 2013 Michael Stapelberg <stapelberg@debian.org>
License: BSD-3-clause

Files: script/service man8/service.8
Copyright: 2006 Red Hat, Inc
2008 Canonical Ltd
License: GPL-2+

Files: script/invoke-rc.d man8/invoke-rc.d.8
Copyright: 2000,2001 Henrique de Moraes Holschuh <hmh@debian.org>
License: GPL-2+

Files: script/update-rc.d man8/update-rc.d.8
Copyright: 1997-2005 Miquel van Smoorenburg <miquels@cistron.nl>
Members of the pkg-sysvinit project
License: GPL-2+

License: GPL-2+
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2, or (at your option)
any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation,
Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

On Debian systems, the complete text of the GNU General Public License
version 2 can be found in /usr/share/common-licenses/GPL-2.
License: BSD-3-clause
Copyright 2013 Michael Stapelberg
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of Michael Stapelberg nor the
  names of contributors may be used to endorse or promote products
  derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY Michael Stapelberg "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL Michael Stapelberg BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.339 simpleclient 0.8.1
1.339.1 Available under license:
No license file was found, but licenses were detected in source scan.

/*
 * Written by Doug Lea with assistance from members of JCP JSR-166
 * Expert Group and released to the public domain, as explained at
 * http://creativecommons.org/publicdomain/zero/1.0/
 * 
 * Source: http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/jsr166e/Striped64.java?revision=1.10
 */

Found in path(s):
* /opt/cola/permits/1262902220_1643563513.99/0/simpleclient-0-8-1-sources-jar/io/prometheus/client/Striped64.java

No license file was found, but licenses were detected in source scan.
/*
* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* http://creativecommons.org/publicdomain/zero/1.0/
*
* Source: http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166e/src/jsr166e/DoubleAdder.java?revision=1.12
*/

/*
Copyright 2012 Andrew Wang (andrew@umbrant.com)

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
*/

/*
Found in path(s):
* /opt/cola/permits/1262902220_1643563513.99/0/simpleclient-0-8-1-sources-
  jar/io/prometheus/client/DoubleAdder.java
No license file was found, but licenses were detected in source scan.
*/

1.340 coreutils 8.28 1ubuntu1

1.340.1 Available under license:

   GNU GENERAL PUBLIC LICENSE
   Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.
The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program—to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must pass on to the recipients the same
freedoms that you received. You must make sure that they, too, receive
or can get the source code. And you must show them these terms so they
know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as
changed, so that their problems will not be attributed erroneously to
authors of previous versions.

Some devices are designed to deny users access to install or run
modified versions of the software inside them, although the manufacturer
can do so. This is fundamentally incompatible with the aim of
protecting users' freedom to change the software. The systematic
pattern of such abuse occurs in the area of products for individuals to
use, which is precisely where it is most unacceptable. Therefore, we
have designed this version of the GPL to prohibit the practice for those
products. If such problems arise substantially in other domains, we
stand ready to extend this provision to those domains in future versions
of the GPL, as needed to protect the freedom of users.
Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the
work under this License, and how to view a copy of this License. If
the interface presents a list of user commands or options, such as a
menu, a prominent item in the list meets this criterion.


The "source code" for a work means the preferred form of the work
for making modifications to it. "Object code" means any non-source
form of a work.

A "Standard Interface" means an interface that either is an official
standard defined by a recognized standards body, or, in the case of
interfaces specified for a particular programming language, one that
is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other
than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major
Component, and (b) serves only to enable use of the work with that
Major Component, or to implement a Standard Interface for which an
implementation is available to the public in source code form. A
"Major Component", in this context, means a major essential component
(kernel, window system, and so on) of the specific operating system
(if any) on which the executable work runs, or a compiler used to
produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all
the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to
control those activities. However, it does not include the work's
System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but
which are not part of the work. For example, Corresponding Source
includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically
linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those
subprograms and other parts of the work.

The Corresponding Source need not include anything that users
can regenerate automatically from other parts of the Corresponding
Source.

The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.
All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.


You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.
You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:
a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation
into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does
not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.


You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a
covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.
In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.
15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.341 commons-lang3 3.4
1.341.1 Available under license:
Apache Commons Lang
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

This product includes software from the Spring Framework, under the Apache License 2.0 (see: StringUtils.containsWhitespace())

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes
'of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.
Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.342 commons-codec 1.11
1.342.1 Available under license:

Apache Commons Codec
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java contains test data from http://aspell.net/test/orig/batch0.tab. Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

===============================================================================
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at http://stevemorse.org/phoneticinfo.htm with permission from the original authors. Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
1.343 jul-to-slf4j 1.7.30

1.343.1 Available under license:

No license file was found, but licenses were detected in source scan.

/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 *
 */

Found in path(s):
* /opt/cola/permits/1135864176_1613618002.1/0/jul-to-slf4j-1-7-30-sources-1-
  jar/org/slf4j/bridge/SLF4JBridgeHandler.java

1.344 libsepol 2.7-1

1.344.1 Available under license:

This is the Debian package for libsepol, and it is built from sources
obtained from http://userspace.selinuxproject.org/releases/current/devel/

This package was debianized by Russell Coker <russell@coker.com.au> on
Fri, 20 Aug 2004 17:26:18 +1000.

libsepol is
This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

On Debian GNU/Linux systems, the complete text of the Lesser GNU General Public License can be found in `/usr/share/common-licenses/LGPL`.

This package is maintained by Manoj Srivastava <srivasta@debian.org>.

The Debian specific changes are 2005-2008, Manoj Srivastava <srivasta@debian.org>, and distributed under the terms of the GNU General Public License, version 2.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL`.

A copy of the GNU General Public License is also available at <URL:http://www.gnu.org/copyleft/gpl.html>. You may also obtain it by writing to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Manoj Srivastava <srivasta@debian.org>
arch-tag: d4250e44-a0e0-4ee0-adb9-2bd74f6eeb27

1.345 mapstruct 1.2.0.Final
1.346 jetty-annotations 9.4.31.v20200723

1.346.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such
Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.
5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided...
that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "["] replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

and
* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

------

Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

------

Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api

------

Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

------

OW2
The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

-----

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

-----

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util
org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----

Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.347 libjpeg 6b

1.347.1 Notifications:
This software is based in part on the work of the Independent JPEG Group.

1.347.2 Available under license:
The Independent JPEG Group's JPEG software

README for release 6b of 27-Mar-1998

This distribution contains the sixth public release of the Independent JPEG Group's free JPEG software. You are welcome to redistribute this software and to use it for any purpose, subject to the conditions under LEGAL ISSUES, below.

Serious users of this software (particularly those incorporating it into larger programs) should contact IJG at jpeg-info@uunet.uu.net to be added to our electronic mailing list. Mailing list members are notified of updates and have a chance to participate in technical discussions, etc.

This software is the work of Tom Lane, Philip Gladstone, Jim Boucher, Lee Crocker, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Guido Vollbeding, Ge’ Weijers, and other members of the Independent JPEG Group.

IJG is not affiliated with the official ISO JPEG standards committee.

DOCUMENTATION ROADMAP

This file contains the following sections:

OVERVIEW General description of JPEG and the IJG software.
LEGAL ISSUES Copyright, lack of warranty, terms of distribution.
REFERENCES Where to learn more about JPEG.
ARCHIVE LOCATIONS Where to find newer versions of this software.
RELATED SOFTWARE Other stuff you should get.
FILE FORMAT WARS  Software *not* to get.
TO DO               Plans for future IJG releases.

Other documentation files in the distribution are:

User documentation:
install.doc     How to configure and install the IJG software.
usage.doc       Usage instructions for cjpeg, djpeg, jpegtran,
                 rdjpgcom, and wrjpgcom.
*.1             Unix-style man pages for programs (same info as usage.doc).
wizard.doc      Advanced usage instructions for JPEG wizards only.
change.log      Version-to-version change highlights.

Programmer and internal documentation:
libjpeg.doc     How to use the JPEG library in your own programs.
example.c       Sample code for calling the JPEG library.
structure.doc   Overview of the JPEG library's internal structure.
filelist.doc    Road map of IJG files.
coderules.doc   Coding style rules --- please read if you contribute code.

Please read at least the files install.doc and usage.doc. Useful information
 can also be found in the JPEG FAQ (Frequently Asked Questions) article. See
ARCHIVE LOCATIONS below to find out where to obtain the FAQ article.

If you want to understand how the JPEG code works, we suggest reading one or
more of the REFERENCES, then looking at the documentation files (in roughly
the order listed) before diving into the code.

OVERVIEW
========

This package contains C software to implement JPEG image compression and
decompression. JPEG (pronounced "jay-peg") is a standardized compression
method for full-color and gray-scale images. JPEG is intended for compressing
"real-world" scenes; line drawings, cartoons and other non-realistic images
are not its strong suit. JPEG is lossy, meaning that the output image is not
exactly identical to the input image. Hence you must not use JPEG if you
have to have identical output bits. However, on typical photographic images,
very good compression levels can be obtained with no visible change, and
remarkably high compression levels are possible if you can tolerate a
low-quality image. For more details, see the references, or just experiment
with various compression settings.

This software implements JPEG baseline, extended-sequential, and progressive
compression processes. Provision is made for supporting all variants of these
processes, although some uncommon parameter settings aren't implemented yet.
For legal reasons, we are not distributing code for the arithmetic-coding
variants of JPEG; see LEGAL ISSUES. We have made no provision for supporting
the hierarchical or lossless processes defined in the standard.

We provide a set of library routines for reading and writing JPEG image files, plus two sample applications "cjpeg" and "djpeg", which use the library to perform conversion between JPEG and some other popular image file formats. The library is intended to be reused in other applications.

In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application. We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.

We welcome the use of this software as a component of commercial products. No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

LEGAL ISSUES
============

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.
This software is copyright (C) 1991-1998, Thomas G. Lane. All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA. ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual distribution terms of the Free Software Foundation; principally, that you must include source code if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot
legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.) So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that
"The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

REFERENCES
==========

We highly recommend reading one or more of these references before trying to understand the innards of the JPEG software.

The best short technical introduction to the JPEG compression algorithm is Wallace, Gregory K. "The JPEG Still Picture Compression Standard", Communications of the ACM, April 1991 (vol. 34 no. 4), pp. 30-44. (Adjacent articles in that issue discuss MPEG motion picture compression, applications of JPEG, and related topics.) If you don't have the CACM issue handy, a PostScript file containing a revised version of Wallace's article is available at ftp://ftp.uu.net/graphics/jpeg/wallace.ps.gz. The file (actually a preprint for an article that appeared in IEEE Trans. Consumer Electronics) omits the sample images that appeared in CACM, but it includes corrections and some added material. Note: the Wallace article is copyright ACM and IEEE, and it may not be used for commercial purposes.

A somewhat less technical, more leisurely introduction to JPEG can be found in "The Data Compression Book" by Mark Nelson and Jean-loup Gailly, published by M&T Books (New York), 2nd ed. 1996, ISBN 1-55851-434-1. This book provides good explanations and example C code for a multitude of compression methods including JPEG. It is an excellent source if you are comfortable reading C code but don't know much about data compression in general. The book's JPEG sample code is far from industrial-strength, but when you are ready to look at a full implementation, you've got one here...

The best full description of JPEG is the textbook "JPEG Still Image Data
The JPEG standard itself is not available electronically; you must order a paper copy through ISO or ITU. (Unless you feel a need to own a certified official copy, we recommend buying the Pennebaker and Mitchell book instead; it's much cheaper and includes a great deal of useful explanatory material.) In the USA, copies of the standard may be ordered from ANSI Sales at (212) 642-4900, or from Global Engineering Documents at (800) 854-7179. (ANSI doesn't take credit card orders, but Global does.) It's not cheap: as of 1992, ANSI was charging $95 for Part 1 and $47 for Part 2, plus 7% shipping/handling. The standard is divided into two parts, Part 1 being the actual specification, while Part 2 covers compliance testing methods. Part 1 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 1: Requirements and guidelines" and has document numbers ISO/IEC IS 10918-1, ITU-T T.81. Part 2 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 2: Compliance testing" and has document numbers ISO/IEC IS 10918-2, ITU-T T.83.

Some extensions to the original JPEG standard are defined in JPEG Part 3, a newer ISO standard numbered ISO/IEC IS 10918-3 and ITU-T T.84. IJG currently does not support any Part 3 extensions.

The JPEG standard does not specify all details of an interchangeable file format. For the omitted details we follow the "JFIF" conventions, revision 1.02. A copy of the JFIF spec is available from:

Literature Department
C-Cube Microsystems, Inc.
1778 McCarthy Blvd.
Milpitas, CA 95035
phone (408) 944-6300, fax (408) 944-6314

A PostScript version of this document is available by FTP at ftp://ftp.uu.net/graphics/jpeg/jfif.ps.gz. There is also a plain text version at ftp://ftp.uu.net/graphics/jpeg/jfif.txt.gz, but it is missing the figures.

The TIFF 6.0 file format specification can be obtained by FTP from ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps.gz. The JPEG incorporation scheme found in the TIFF 6.0 spec of 3-June-92 has a number of serious problems. IJG does not recommend use of the TIFF 6.0 design (TIFF Compression tag 6). Instead, we recommend the JPEG design proposed by TIFF Technical Note #2 (Compression tag 7). Copies of this Note can be obtained from ftp.sgi.com or from ftp://ftp.uu.net/graphics/jpeg/. It is expected that the next revision of the TIFF spec will replace the 6.0 JPEG design with the Note's design. Although IJG's own code does not support TIFF/JPEG, the free libtiff library
uses our library to implement TIFF/JPEG per the Note. libtiff is available from ftp://ftp.sgi.com/graphics/tiff/.

ARCHIVE LOCATIONS
=================

The "official" archive site for this software is ftp.uu.net (Internet address 192.48.96.9). The most recent released version can always be found there in directory graphics/jpeg. This particular version will be archived as ftp://ftp.uu.net/graphics/jpeg/jpegsrc.v6b.tar.gz. If you don't have direct Internet access, UUNET's archives are also available via UUCP; contact help@uunet.uu.net for information on retrieving files that way.

Numerous Internet sites maintain copies of the UUNET files. However, only ftp.uu.net is guaranteed to have the latest official version.

You can also obtain this software in DOS-compatible "zip" archive format from the SimTel archives (ftp://ftp.simtel.net/pub/simtelnet/msdos/graphics/), or on CompuServe in the Graphics Support forum (GO CIS:GRAPHSUP), library 12 "JPEG Tools". Again, these versions may sometimes lag behind the ftp.uu.net release.

The JPEG FAQ (Frequently Asked Questions) article is a useful source of general information about JPEG. It is updated constantly and therefore is not included in this distribution. The FAQ is posted every two weeks to Usenet newsgroups comp.graphics.misc, news.answers, and other groups. It is available on the World Wide Web at http://www.faqs.org/faqs/jpeg-faq/ and other news.answers archive sites, including the official news.answers archive at rtfm.mit.edu: ftp://rtfm.mit.edu/pub/usenet/news.answers/jpeg-faq/.

If you don't have Web or FTP access, send e-mail to mail-server@rtfm.mit.edu with body
send usenet/news.answers/jpeg-faq/part1
send usenet/news.answers/jpeg-faq/part2

RELATED SOFTWARE
=================

Numerous viewing and image manipulation programs now support JPEG. (Quite a few of them use this library to do so.) The JPEG FAQ described above lists some of the more popular free and shareware viewers, and tells where to obtain them on Internet.

If you are on a Unix machine, we highly recommend Jef Poskanzer's free PBMPLUS software, which provides many useful operations on PPM-format image files. In particular, it can convert PPM images to and from a wide range of other formats, thus making cjpeg/djpeg considerably more useful. The latest
version is distributed by the NetPBM group, and is available from numerous sites, notably ftp://wuarchive.wustl.edu/graphics/graphics/packages/NetPBM/. Unfortunately PBMPPLUS/NETPBM is not nearly as portable as the IJG software is; you are likely to have difficulty making it work on any non-Unix machine.

A different free JPEG implementation, written by the PVRG group at Stanford, is available from ftp://havefun.stanford.edu/pub/jpeg/. This program is designed for research and experimentation rather than production use; it is slower, harder to use, and less portable than the IJG code, but it is easier to read and modify. Also, the PVRG code supports lossless JPEG, which we do not. (On the other hand, it doesn't do progressive JPEG.)

FILE FORMAT WARS
================

Some JPEG programs produce files that are not compatible with our library. The root of the problem is that the ISO JPEG committee failed to specify a concrete file format. Some vendors "filled in the blanks" on their own, creating proprietary formats that no one else could read. (For example, none of the early commercial JPEG implementations for the Macintosh were able to exchange compressed files.)

The file format we have adopted is called JFIF (see REFERENCES). This format has been agreed to by a number of major commercial JPEG vendors, and it has become the de facto standard. JFIF is a minimal or "low end" representation. We recommend the use of TIFF/JPEG (TIFF revision 6.0 as modified by TIFF Technical Note #2) for "high end" applications that need to record a lot of additional data about an image. TIFF/JPEG is fairly new and not yet widely supported, unfortunately.

The upcoming JPEG Part 3 standard defines a file format called SPIFF. SPIFF is interoperable with JFIF, in the sense that most JFIF decoders should be able to read the most common variant of SPIFF. SPIFF has some technical advantages over JFIF, but its major claim to fame is simply that it is an official standard rather than an informal one. At this point it is unclear whether SPIFF will supersede JFIF or whether JFIF will remain the de-facto standard. IJG intends to support SPIFF once the standard is frozen, but we have not decided whether it should become our default output format or not. (In any case, our decoder will remain capable of reading JFIF indefinitely.)

Various proprietary file formats incorporating JPEG compression also exist. We have little or no sympathy for the existence of these formats. Indeed, one of the original reasons for developing this free software was to help force convergence on common, open format standards for JPEG files. Don't use a proprietary file format!
TO DO
=====

The major thrust for v7 will probably be improvement of visual quality. The current method for scaling the quantization tables is known not to be very good at low Q values. We also intend to investigate block boundary smoothing, "poor man's variable quantization", and other means of improving quality-vs-file-size performance without sacrificing compatibility.

In future versions, we are considering supporting some of the upcoming JPEG Part 3 extensions --- principally, variable quantization and the SPIFF file format.

As always, speeding things up is of great interest.

Please send bug reports, offers of help, etc. to jpeg-info@uunet.uu.net.

1.348 httpcomponents-client 4.5.13
1.348.1 Available under license:

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

========================================================================

This project includes Public Suffix List copied from <https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==================================

1. Definitions
---------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
    means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.

1.4. "Covered Software"
    means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
    means
    (a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or
(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this
License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party’s modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:
(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

-----------------------------------------------
If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is" basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a
7. Limitation of Liability

Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License
---------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.
You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.349 shiro-web 1.7.1

1.349.1 Available under license:

Apache Shiro
Copyright 2008-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

The implementation for org.apache.shiro.util.SoftHashMap is based
on initial ideas from Dr. Heinz Kabutz’s publicly posted version
available at http://www.javaspecialists.eu/archive/Issue015.html,
with continued modifications.

Certain parts (StringUtils, IpAddressMatcher, etc.) of the source
code for this product was copied for simplicity and to reduce
dependencies from the source code developed by the Spring Framework
Project (http://www.springframework.org).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
   Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
   stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
   that You distribute, all copyright, patent, trademark, and
   attribution notices from the Source form of the Work,
   excluding those notices that do not pertain to any part of
   the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.350 j2objc-annotations 1.3

1.351 log4j-web 2.17.0
1.351.1 Available under license:

Apache Log4j Web
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.352 jetty-xml 9.4.31.v20200723

1.352.1 Available under license:

This program and the accompanying materials are made available under the terms of the Eclipse Public License 2.0 which is available at http://www.eclipse.org/legal/epl-2.0, or the Apache Software License 2.0 which is available at https://www.apache.org/licenses/LICENSE-2.0.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:
   i) changes to the Program, and
   ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor
itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.
3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
   i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
   ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
   iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
   iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (“Commercial Contributor”) hereby agrees to defend and indemnify every other Contributor (“Indemnified Contributor”) against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims
or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

a) promptly notify the Commercial Contributor in writing of such claim, and
b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent
necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
this License, each Contributor hereby grants to You a perpetual,
worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You
institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties of TITLE, NONINFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====================================================================
Jetty Web Container
Copyright 1995-2018 Mort Bay Consulting Pty Ltd.
=====================================================================

The Jetty Web Container is Copyright Mort Bay Consulting Pty Ltd unless otherwise noted.

Jetty is dual licensed under both

* The Apache 2.0 License
  http://www.apache.org/licenses/LICENSE-2.0.html

  and

* The Eclipse Public 1.0 License

Jetty may be distributed under either license.

-----
Eclipse

The following artifacts are EPL.
* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following artifacts are EPL and ASL2.
* org.eclipse.jetty.orbit:javax.security.auth.message

The following artifacts are EPL and CDDL 1.0.
* org.eclipse.jetty.orbit:javax.mail.glassfish

-----
Oracle

The following artifacts are CDDL + GPLv2 with classpath exception.
https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

* javax.servlet:javax.servlet-api
* javax.annotation:javax.annotation-api
* javax.transaction:javax.transaction-api
* javax.websocket:javax.websocket-api
Oracle OpenJDK

If ALPN is used to negotiate HTTP/2 connections, then the following artifacts may be included in the distribution or downloaded when ALPN module is selected.

* java.sun.security.ssl

These artifacts replace/modify OpenJDK classes. The modifications are hosted at github and both modified and original are under GPL v2 with classpath exceptions.
http://openjdk.java.net/legal/gplv2+ce.html

OW2

The following artifacts are licensed by the OW2 Foundation according to the terms of http://asm.ow2.org/license.html

org.ow2.asm:asm-commons
org.ow2.asm:asm

Apache

The following artifacts are ASL2 licensed.

org.apache.taglibs:taglibs-standard-spec
org.apache.taglibs:taglibs-standard-impl

MortBay

The following artifacts are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

org.mortbay.jasper:apache-jsp
org.apache.tomcat:tomcat-jasper
org.apache.tomcat:tomcat-juli
org.apache.tomcat:tomcat-jsp-api
org.apache.tomcat:tomcat-el-api
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-api
org.apache.tomcat:tomcat-util-scan
org.apache.tomcat:tomcat-util

org.mortbay.jasper:apache-el
org.apache.tomcat:tomcat-jasper-el
org.apache.tomcat:tomcat-el-api

-----
Mortbay

The following artifacts are CDDL + GPLv2 with classpath exception.

https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

org.eclipse.jetty.toolchain:jetty-schemas

-----
Assorted

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville. Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

1.353 scowl 2017.08.24-1
1.353.1 Available under license:

This package was debianized by David Coe <davide@debian.org> on Sun, 15 Jun 2003.

SCOWL (Spell Checker Oriented Word Lists) is a collection of English word lists maintained by Kevin Atkinson <kevina@users.sourceforge.net>

It was downloaded from:
http://prdownloads.sourceforge.net/wordlist/scowl-5.tar.gz

SF project page:
http://wordlist.sourceforge.net/

Copyright: (extracted from the SCOWL README file):

The collective work is Copyright 2000-2011 by Kevin Atkinson as well as any of the copyrights mentioned below:
Copyright 2000-2011 by Kevin Atkinson

Permission to use, copy, modify, distribute and sell these word lists, the associated scripts, the output created from the scripts, and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Kevin Atkinson makes no representations about the suitability of this array for any purpose. It is provided “as is” without express or implied warranty.

Alan Beale <biljir@pobox.com> also deserves special credit as he has, in addition to providing the 12Dicts package and being a major contributor to the ENABLE word list, given me an incredible amount of feedback and created a number of special lists (those found in the Supplement) in order to help improve the overall quality of SCOWL.

The 10 level includes the 1000 most common English words (according to the Moby (TM) Words II [MWords] package), a subset of the 1000 most common words on the Internet (again, according to Moby Words II), and frequently class 16 from Brian Kelk’s "UK English Wordlist with Frequency Classification”.

The MWords package was explicitly placed in the public domain:

The Moby lexicon project is complete and has been place into the public domain. Use, sell, rework, excerpt and use in any way on any platform.

Placing this material on internal or public servers is also encouraged. The compiler is not aware of any export restrictions so freely distribute world-wide.

You can verify the public domain status by contacting

Grady Ward
3449 Martha Ct.
Arcata, CA  95521-4884

grady@netcom.com
grady@northcoast.com

The "UK English Wordlist With Frequency Classification” is also in the Public Domain:

Date: Sat, 08 Jul 2000 20:27:21 +0100
From: Brian Kelk <Brian.Kelk@cl.cam.ac.uk>
I was wondering what the copyright status of your "UK English Wordlist With Frequency Classification" word list as it seems to be lacking any copyright notice.

There were many, many sources in total, but any text marked "copyright" was avoided. Locally-written documentation was one source. An earlier version of the list resided in a filespace called PUBLIC on the University mainframe, because it was considered public domain.

Date: Tue, 11 Jul 2000 19:31:34 +0100

So are you saying your word list is also in the public domain?

That is the intention.

The 20 level includes frequency classes 7-15 from Brian's word list.

The 35 level includes frequency classes 2-6 and words appearing in at least 11 of 12 dictionaries as indicated in the 12Dicts package. All words from the 12Dicts package have had likely inflections added via my inflection database.

The 12Dicts package and Supplement is in the Public Domain.

The WordNet database, which was used in the creation of the Inflections database, is under the following copyright:

This software and database is being provided to you, the LICENSEE, by Princeton University under the following license. By obtaining, using and/or copying this software and database, you agree that you have read, understood, and will comply with these terms and conditions:

Permission to use, copy, modify and distribute this software and database and its documentation for any purpose and without fee or royalty is hereby granted, provided that you agree to comply with the following copyright notice and statements, including the disclaimer, and that the same appear on ALL copies of the software, database and documentation, including modifications that you make for internal use or for distribution.

WordNet 1.6 Copyright 1997 by Princeton University. All rights reserved.

THIS SOFTWARE AND DATABASE IS PROVIDED "AS IS" AND PRINCETON UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PRINCETON UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE LICENSED SOFTWARE, DATABASE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

The name of Princeton University or Princeton may not be used in advertising or publicity pertaining to distribution of the software and/or database. Title to copyright in this software, database and any associated documentation shall at all times remain with Princeton University and LICENSEEE agrees to preserve same.

The 40 level includes words from Alan's 3esl list found in version 4.0 of his 12dicts package. Like his other stuff the 3esl list is also in the public domain.

The 50 level includes Brian's frequency class 1, words words appearing in at least 5 of 12 of the dictionaries as indicated in the 12Dicts package, and uppercase words in at least 4 of the previous 12 dictionaries. A decent number of proper names is also included: The top 1000 male, female, and Last names from the 1990 Census report; a list of names sent to me by Alan Beale; and a few names that I added myself. Finally a small list of abbreviations not commonly found in other word lists is included.

The name files form the Census report is a government document which I don't think can be copyrighted.

The file special-jargon.50 uses common.lst and word.lst from the "Unofficial Jargon File Word Lists" which is derived from "The Jargon File". All of which is in the Public Domain. This file also contain a few extra UNIX terms which are found in the file "unix-terms" in the special/ directory.

The 55 level includes words from Alan's 2of4brif list found in version 4.0 of his 12dicts package. Like his other stuff the 2of4brif is also in the public domain.

The 60 level includes all words appearing in at least 2 of the 12 dictionaries as indicated by the 12Dicts package.

The 70 level includes Brian's frequency class 0 and the 74,550 common dictionary words from the MWords package. The common dictionary words, like those from the 12Dicts package, have had all likely inflections added. The 70 level also included the 5desk list from version 4.0 of the 12Dics package which is the public domain.

The 80 level includes the ENABLE word list, all the lists in the
ENABLE supplement package (except for ABLE), the "UK Advanced Cryptics Dictionary" (UKACD), the list of signature words in from YAWL package, and the 10,196 places list from the MWords package.

The ENABLE package, maintained by M\Cooper <thegrendel@theriver.com>, is in the Public Domain:

The ENABLE master word list, WORD.LST, is herewith formally released into the Public Domain. Anyone is free to use it or distribute it in any manner they see fit. No fee or registration is required for its use nor are "contributions" solicited (if you feel you absolutely must contribute something for your own peace of mind, the authors of the ENABLE list ask that you make a donation on their behalf to your favorite charity). This word list is our gift to the Scrabble community, as an alternate to "official" word lists. Game designers may feel free to incorporate the WORD.LST into their games. Please mention the source and credit us as originators of the list. Note that if you, as a game designer, use the WORD.LST in your product, you may still copyright and protect your product, but you may *not* legally copyright or in any way restrict redistribution of the WORD.LST portion of your product. This *may* under law restrict your rights to restrict your users' rights, but that is only fair.

UKACD, by J Ross Beresford <ross@bryson.demon.co.uk>, is under the following copyright:

Copyright (c) J Ross Beresford 1993-1999. All Rights Reserved.

The following restriction is placed on the use of this publication: if The UK Advanced Cryptics Dictionary is used in a software package or redistributed in any form, the copyright notice must be prominently displayed and the text of this document must be included verbatim.

There are no other restrictions: I would like to see the list distributed as widely as possible.

The 95 level includes the 354,984 single words, 256,772 compound words, 4,946 female names and the 3,897 male names, and 21,986 names from the MWords package, ABLE.LST from the ENABLE Supplement, and some additional words found in my part-of-speech database that were not found anywhere else.

Accent information was taken from UKACD.

My VARCON package was used to create the American, British, and Canadian word list.
Since the original word lists used in the VARCON package came from the Ispell distribution they are under the Ispell copyright:

Copyright 1993, Geoff Kuenning, Granada Hills, CA
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All modifications to the source code must be clearly marked as such. Binary redistributions based on modified source code must be clearly marked as modified versions in the documentation and/or other materials provided with the distribution.
   (clause 4 removed with permission from Geoff Kuenning)
4. The name of Geoff Kuenning may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY GEOFF KUENNING AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL GEOFF KUENNING OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.354 libsigar 1.6.2
1.354.1 Available under license:
Hyperic SIGAR includes some third-party open source components in its distribution. The list below identifies the community or organization and links to their appropriate license terms.

The Hyperic team would like to thank all the communities of the projects listed below for their contributions.
The following components are included without modification:

- log4j -
Information: http://logging.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

- junit -
Information: http://www.junit.org/index.htm
License: http://www.opensource.org/licenses/cpl.php

The following components are included with modification:

- cpptasks -
Information: http://ant-contrib.sourceforge.net/
License: http://ant-contrib.sourceforge.net/LICENSE.txt

- (portions of) APR -
Information: http://apr.apache.org/
License: http://www.apache.org/licenses/LICENSE-2.0

- solaris get_mib2 -
Information: ftp://vic.cc.purdue.edu/pub/tools/unix/solaris/get_mib2/
License: within src/os/solaris/get_mib2.[ch]

- getline by Chris Thewalt -
Information: http://tinyurl.com/r438r
License: within src/sigar_getline.c

- PrintfFormat.java -
Information: http://java.sun.com/developer/technicalArticles/Programming/sprintf/PrintfFormat.java
License: within bindings/java/src/org/hyperic/sigar/util/PrintfFormat.java
This product includes software developed by
  The Ant-Contrib project (http://ant-contrib.sourceforge.net)
This product includes software developed by
  The Apache Software Foundation (http://www.apache.org/).

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation’s software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author’s protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors’ reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's
source code as you receive it, in any medium, provided that you
conspicuously and appropriately publish on each copy an appropriate
copyright notice and disclaimer of warranty; keep intact all the
notices that refer to this License and to the absence of any warranty;
and give any other recipients of the Program a copy of this License
along with the Program.

You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices
      stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in
      whole or in part contains or is derived from the Program or any
      part thereof, to be licensed as a whole at no charge to all third
      parties under the terms of this License.

   c) If the modified program normally reads commands interactively
      when run, you must cause it, when started running for such
      interactive use in the most ordinary way, to print or display an
      announcement including an appropriate copyright notice and a
      notice that there is no warranty (or else, saying that you provide
      a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source
code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, 
but WITHOUT ANY WARRANTY; without even the implied warranty of 
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the 
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along 
with this program; if not, write to the Free Software Foundation, Inc., 
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this 
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author 
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. 
This is free software, and you are welcome to redistribute it 
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate 
parts of the General Public License. Of course, the commands you use may 
be called something other than `show w' and `show c'; they could even be 
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your 
school, if any, to sign a "copyright disclaimer" for the program, if 
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989 
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into 
proprietary programs. If your program is a subroutine library, you may 
consider it more useful to permit linking proprietary applications with the 
library. If this is what you want to do, use the GNU Lesser General 
Public License instead of this License.

Apache License 
Version 2.0, January 2004 
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement You may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
extcept as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and
1.355.1 Available under license:

Unless otherwise *explicitly* stated the following text describes the licensed conditions under which the contents of this libpamc release may be distributed:

Redistribution and use in source and binary forms of libpamc, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain any existing copyright notice, and this entire permission notice in its entirety, including the disclaimer of warranties.

2. Redistributions in binary form must reproduce all prior and current copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of any author may not be used to endorse or promote products derived from this software without their specific prior written permission.

ALTERNATIVELY, this product may be distributed under the terms of the GNU Library General Public License (LGPL), in which case the provisions of the GNU LGPL are required INSTEAD OF the above restrictions. (This clause is necessary due to a potential conflict between the GNU LGPL and the restrictions contained in a BSD-style copyright.)

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Unless otherwise *explicitly* stated the following text describes the licensed conditions under which the contents of this Linux-PAM release may be distributed:

-------------------------------------------------------------------------
Redistribution and use in source and binary forms of Linux-PAM, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain any existing copyright notice, and this entire permission notice in its entirety, including the disclaimer of warranties.

2. Redistributions in binary form must reproduce all prior and current copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of any author may not be used to endorse or promote products derived from this software without their specific prior written permission.

ALTERNATIVELY, this product may be distributed under the terms of the GNU General Public License, in which case the provisions of the GNU GPL are required INSTEAD OF the above restrictions. (This clause is necessary due to a potential conflict between the GNU GPL and the restrictions contained in a BSD-style copyright.)

THIS SOFTWARE IS PROVIDED ""AS IS"" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-------------------------------------------------------------------------
Patch for Ubuntu bug #399071

Display the contents of /etc/legal as part of the MOTD, the first time the user logs in, and set a flag in the user's homedir if possible to prevent repeat displays.

Authors: Dustin Kirkland <kirkland@canonical.com>
Upstream status: Ubuntu-specific, maybe submit to Debian

Index: pam.ubuntu/modules/pam_motd/pam_motd.c

--- pam.ubuntu.orig/modules/pam_motd/pam_motd.c
+++ pam.ubuntu/modules/pam_motd/pam_motd.c
@@ -73,6 +73,61 @@
close(fd);
}

+int display_legal(pam_handle_t *pamh)
+{
+   int retval = PAM_IGNORE, rc;
+   char *user = NULL;
+   char *dir = NULL;
+   char *flag = NULL;
+   struct passwd *pwd = NULL;
+   struct stat s;
+   int f;
+   /* Get the user name to determine if we need to print the disclaimer */
+   rc = pam_get_item(pamh, PAM_USER, &user);
+   if (rc == PAM_SUCCESS && user != NULL && *(const char *)user != '\0')
+   {
+      PAM_MODUTIL_DEF_PRIVS(privs);
+      /* Get the password entry */
+      pwd = pam_modutil_getpwnam (pamh, user);
+      if (pwd != NULL)
+      {
+         if (pam_modutil_drop_priv(pamh, &privs, pwd))
+            pam_syslog(pamh, LOG_ERR,
+                       "Unable to change UID to %d temporarily\n",
+                       pwd->pw_uid);
+         retval = PAM_SESSION_ERR;
+         goto finished;
+      }
+      if (asprintf(&dir, "%s/.cache", pwd->pw_dir) == -1 || !dir)
+         goto finished;
+      if (asprintf(&flag, "%s/motd.legal-displayed", dir) == -1 || !flag)
+         goto finished;
+      if (stat(flag, &s) != 0)
+      {
+         display_file(pamh, "/etc/legal");
+         mkdir(dir, 0700);
+         f = open(flag, O_WRONLY|O_CREAT|O_EXCL,
+                  S_IRUSR|S_IWUSR|S_IRGRP|S_IROTH);
    if (f>=0) close(f);
+
+}  
+
+finished:
+    if (pam_modutil_regain_priv(pamh, &privs)) {
+        pam_syslog(pamh, LOG_ERR,
+                   "Unable to change UID back to %d\n", privs.old_uid);
+        retval = PAM_SESSION_ERR;
+    }
+
+    _pam_drop(flag);
+    _pam_drop(dir);
+}  
+
+    return retval;
+
+}
+
+PAM_EXTERN
+int pam_sm_open_session(pam_handle_t *pamh, int flags,
+                        int argc, const char **argv)
@@ -116,6 +171,9 @@

    /* Display the updated motd */
    display_file(pamh, motd_path);

+    /* Display the legal disclaimer only if necessary */
+    retval = display_legal(pamh);
+    return retval;
+
}  

This package was debianized by J.H.M. Dassen (Ray) jdassen@debian.org on

It was downloaded from ftp://ftp.kernel.org/pub/linux/libs/pam/pre/

Copyright (C) 1994, 1995, 1996 Olaf Kirch, <okir@monad.swb.de>
Copyright (C) 1995 Wietse Venema
Copyright (C) 1995, 2001-2008 Red Hat, Inc.
Copyright (C) 1996-1999, 2000-2003, 2005 Andrew G. Morgan <morgan@kernel.org>
Copyright (C) 1996, 1997, 1999 Cristian Gafton <gafton@redhat.com>
Copyright (C) 1996, 1999 Theodore Ts'o
Copyright (C) 1996 Alexander O. Yuriev
Copyright (C) 1996 Elliot Lee
Copyright (C) 1997 Philip W. Dalrymple <pwd@mdtsoft.com>
Copyright (C) 1999 Jan Rkorajski
Copyright (C) 1999 Ben Collins <bcollins@debian.org>
Copyright (C) 2003, 2005 IBM Corporation
Copyright (C) 2003, 2006 SuSE Linux AG.
Unless otherwise *explicitly* stated the following text describes the licensed conditions under which the contents of this Linux-PAM release may be distributed:

Redistribution and use in source and binary forms of Linux-PAM, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain any existing copyright notice, and this entire permission notice in its entirety, including the disclaimer of warranties.

2. Redistributions in binary form must reproduce all prior and current copyright notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of any author may not be used to endorse or promote products derived from this software without their specific prior written permission.

ALTERNATIVELY, this product may be distributed under the terms of the GNU General Public License, in which case the provisions of the GNU GPL are required INSTEAD OF the above restrictions. (This clause is necessary due to a potential conflict between the GNU GPL and the restrictions contained in a BSD-style copyright.)

THIS SOFTWARE IS PROVIDED `AS IS` AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

On Debian GNU/Linux systems, the complete text of the GNU General Public License can be found in `/usr/share/common-licenses/GPL`.
1.356 apache-log4j 2.17.0

1.356.1 Available under license:

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.357 commons-net 3.4

1.357.1 Available under license:

Apache Commons Net
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License);
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.358 jboss-logging 3.2.0.Final
1.358.1 Available under license:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.359 attr 2.4.47-2build1

1.359.1 Available under license :
This package was debianized by Nathan Scott nathans@debian.org on
Sun, 19 Nov 2000 07:37:09 -0500.

It can be downloaded from ftp://oss.sgi.com/projects/xfs/download/

Copyright:

Copyright (C) 2001-2002 Silicon Graphics, Inc. All Rights Reserved.
Copyright (C) 2001 Andreas Gruenbacher.

You are free to distribute this software under Version 2.1
of the GNU Lesser General Public License.
On Debian systems, refer to /usr/share/common-licenses/LGPL-2.1
for the complete text of the GNU Lesser General Public License.

Certain components (as annotated in the source) are licensed
under version 2 of the terms of the GNU General Public License.
On Debian systems, the complete text of the GNU General Public
License can be found in /usr/share/common-licenses/GPL file.
Most components of the "attr" package are licensed under
Version 2.1 of the GNU Lesser General Public License (see below).
below.

Some components (as annotated in the source) are licensed
under Version 2 of the GNU General Public License (see COPYING).

-------------------------------------------------------------------

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]
there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU
operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.
1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of...
this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.
However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified
executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients’ exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply,
and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY
15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.
You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Most components of the "attr" package are licensed under Version 2.1 of the GNU Lesser General Public License (see COPYING.LGPL).

Some components (as annotated in the source) are licensed under Version 2 of the GNU General Public License (see below),

---------------------------------------------------

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, so
that any problems introduced by others will not reflect on the original
authors' reputations.

Finally, any free program is threatened constantly by software
patents. We wish to avoid the danger that redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and
modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification"). Each licensee is addressed as "you".
Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.
It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices. Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded. In such case, this License incorporates
the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation. If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.
If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.